



Near Field Communication Tag Management System: TagMan case study

Alexei Chugunov

Master's thesis
May 2013
Information Technology

ABSTRACT

Tampere University of Applied Sciences
Degree Programme in Information Technology

Alexei Chugunov:
Near Field Communication Tag Management System: TagMan case study

Thesis supervisor Mikkolainen Jari

Master's thesis 65 pages, appendices 9 pages
May 2013

Near Field Communication is relatively emerging technology which will change many aspects of everyday life. There are a lot of use cases in payment and ticketing, infotainment services and field force solutions where NFC could be utilized. NFC technology grown from RFID and includes many RFID standards, including standards for NFC tags. In the near future we will see a lot of NFC tags at streets, homes and offices, which will require tag management system to handle them. The aim of this thesis was to research different tag management models and approaches and to develop NFC tag management system in order to maintain and control NFC tag ecosystem.

In this work tag management system was analyzed from many prospective, including technical, business and security aspects. Possibility of integration tag management system into asset management, field force solutions and location based services was discussed. One tag many service concept was introduced, allowing to use same tag for different users and services. Different types of identifiers such as barcode, EPC code, QR code, which could be used in tag management system along with NFC tags, was compared to reveal its benefits and drawbacks.

During this thesis work prototype of NFC tag management system named TagMan was developed for ToP Tunniste, company located at Tampere, which many years develop NFC and RFID solutions. TagMan consists of web pages for managing tags, tag groups, locations and schedules. To analyze usage of NFC tags the statistic of tag taps is available. TagMan also includes mobile application for Android to encode TagMan NFC tags and insert them into database. Further development of system and its full deployment is next step.

Key words: near field communication, tag management system, NFC tags

FOREWORD

I would like to express my gratitude to Pauli Tossavainen, CEO of ToP Tunniste and all my colleagues who helped me with ideas and supported me during my work. Also I would like to thank Mr. Jari Mikkolainen for supervising this thesis work.

Finally I would like to thank my wife, Natalia, for supporting me and my son Nikita, who was born year 2011, for unforgettable memories and happiness brought to my life.

Tampere, April 06, 2012

Alexei Chugunov.

CONTENTS

1	INTRODUCTION	7
2	Near Field Communication	8
2.1	NFC overview	8
2.1.1	Radio Frequency Identification	9
2.1.2	From RFID to NFC	10
2.1.3	Internet of things and physical browsing	10
2.1.4	NFC use cases	11
2.2	NFC architecture	13
2.2.1	Three modes of NFC	15
2.2.2	NFC Data Exchange Format	18
2.2.3	Record Type Definition	19
2.2.4	NFC Forum tag types	21
2.3	NFC devices and NFC development	22
2.3.1	NFC Development	22
2.3.2	NFC devices overview	23
2.3.3	NFC mobile device architecture	24
2.4	NFC security aspects	26
3	NFC Tag management and audit	29
3.1	Smart Poster and Tag emulation	29
3.1.1	Smart Poster	29
3.1.2	Tag Emulation with NFC reader and mobile phone	30
3.2	Tag management	31
3.2.1	Models and Concepts	33
3.2.2	Business aspects	35
3.2.3	Security and other issues	36
3.3	Asset management with NFC	38
4	Case study: TagMan - NFC Tag management system	40
4.1	TagMan overview	40
4.2	TagMan Use cases	40
4.3	Development environment	42
4.4	TagMan architecture	44
4.4.1	Database implementation	45
4.4.2	Server-side architecture	47
4.4.3	Mobile application	48
4.4.4	Web interface	52
4.4.5	TagMan NFC Tag	62

4.5 Future Development	63
5 Results and conclusions.....	64
6 Summary	65
REFERENCES.....	66
APPENDICES	69
Appendix 1. Title.....	69
Appendix 2. Title.....	77

ABBREVIATIONS AND TERMS

TAMK	Tampere University of Applied Sciences
NFC	Near Field Communication
RTD	Record Type definition
DB	Database
JavaScript	Scripting language
PHP	Server-side scripting language
APACHE	HTTP Server software
API	Application programming interface
SQL	Sequential Query Language
HTML	Hypertext Markup Language
HTTP	Hypertext Transport Protocol
MySQL	Database Server
URI	Unified Relation Identifier
URL	Uniform Resource Locator (this is a special case of an URI)
SP	Smart Poster, type of RTD
RFID	Radio Frequency Identification
Auto-ID	Automatic identification
NDEF	NFC Data Exchange Format
SMS	Short Message Service
ISO	International Standard Organisation
JSR	Java Specification Request
LLCP	Logical Link Control Protocol
TNF	Type Name Field, field in NDEF message
UICC	Universal Integrated Circuit Card
EPC	Electronic Product Code
QR code	Visual 2D barcode
GPS	Global Positioning System
MNO	Mobile Network Operator
Qt	Cross-platform application framework
SWP	Single Wire protocol
SDK	Software development kit
SE	Secure element

1 INTRODUCTION

NFC is relatively new technology which starts to influence daily life of millions of people worldwide. It creates great opportunity to companies to create new innovative products and services. The purpose of NFC is to make life easier and improve usability of different aspects of life such as payment, ticketing, data transfer between different devices.

In recent years millions of NFC devices such as NFC mobile phones, desktop readers, NFC tags were manufactured and came to people's life. Those devices and service are creating huge NFC ecosystem. One of important part of NFC ecosystem is NFC tags. Billions of tags are already in use in ticketing and access control systems. The more of them will be manufactured in the nearest future.

NFC tags could contain different types of data. It could have URL link to web page, phone number, SMS message, VCard with personal contact information, text, other proprietary data to be used by mobile phone or desktop application. Data on tag could be protected or rewritable. Billions of tags will appear at streets, shops, company's premises, homes. Such big amount of tags requires a system which will manage those tags.

The purpose of my thesis work is to create and analyze NFC tag management system. Tag management system will handle NFC tag dynamically, changing content of tag based on different aspects. For example time, content of tag will be changed depending weather it is morning, afternoon, evening or night. Also it would be possible to modify tag content based on day of the week. As we see dynamically generated content create big opportunities for advertisement and other infotainment services. TagMan system is a tool which provides convenient way to manage NFC tag ecosystem.

2 Near Field Communication

NFC, short for Near Field Communication, is a short range wireless RFID technology that makes use of interacting electromagnetic radio fields instead of the typical direct radio transmissions used by technologies such as Bluetooth. It is meant for applications where a physical touch, or close to it, is required in order to maintain security.

2.1 NFC overview

Near Field Communication (NFC) is a currently emerging and yet promising area which will have an enormous impact on the financial ecosystem as well as mobile technology throughout the world within just a few years.

NFC, being a short range wireless communication technology that potentially facilitates mobile phone usage of billions of people throughout the world offers an enormous number of use cases including credit cards, debit cards, loyalty cards, car keys, access keys for hotels, offices and houses, field force solutions, applications eventually integrating all such materials into one single mobile phone.

Payment and ticketing have been considered as ‘killer applications’ of NFC. One of the main advantages of NFC it can simulate a smart card. The NFC is built as such that it is backward compatible with the contactless card standards.

Near Field Communication (NFC) brings a new twist for the mobile payment as it can embed the functions that we have gotten used with different contactless cards like credit cards, public transportation tickets and alike. Combination of mobile devices and NFC opens up new possibilities for mobile payment, which as a whole has encountered different problems and its volume has been small. As a new technology that needs an infrastructure to work, the process of adaptation has been slow when compared to other technologies integrated to mobile devices, for example camera. Secure payment and ticketing function needs different kind of architecture than straight NFC-function with reader and tag. Some kind of secure chip must be included into the architecture for the secure place for persistent data.

2.1.1 Radio Frequency Identification

Radio frequency identification (RFID) technology uses radio waves to automatically identify physical objects. Thus, RFID is an example of automatic identification (Auto-ID) technology by which a physical object can be identified automatically. Other examples of Auto-ID include bar code, biometric (for example, using fingerprint), voice identification, and optical character recognition (OCR) systems.

The concept of automatic identification using a radio transponder originated in World War II as a way to distinguish friendly aircraft from the enemy; hence, the name Identification Friend or Foe (IFF). The “friendly” planes responded with the correct identification, while those that did not respond were considered “foes.”

In principle, IFF operates much the same as RFID. A coded interrogation signal is sent out on a particular RF, which the transponder receives and decodes. The transponder then replies with encrypted identification information.

In a nutshell, RFID involves detecting and identifying a tagged object through the data it transmits. This requires a tag, a reader and antenna located at each end of the system. The reader is typically connected to a host computer or other device that has the necessary intelligence to further process the tag data and take action. One key element of operation in RFID is data transfer. It occurs with the connection between a tag and a reader, also known as coupling. The coupling in most RFID systems is either electromagnetic (backscatter) or magnetic (inductive). The element that enables the tag and reader communication is the antenna. The tag and the reader each have its own antenna.

Another important element in an RFID system is the frequency of operation between the tag and the reader. Specific frequency selection is driven by application requirements such as speed, accuracy, and environmental conditions, with standards and regulations that govern specific applications. For example, RFID applications for animal tagging have been operating in the 135 kHz frequency band, based on longstanding regulations and accepted standards.

2.1.2 From RFID to NFC

The NFC concept is designed from the synergy of several technologies including wireless communications, mobile devices, mobile applications and smart cards. So what is genesis of NFC, why the need for such technology has arisen? The changes or improvements on RFID to expose NFC technology can be described below. First of all NFC is short range communication, which is good for secure and intuitive interaction and RFID may use long range especially for active tags that contain embedded energy. NFC uses only passive tags, that simplifies deployment of NFC system and RFID uses both active and passive tags. NFC utilizes secure data exchange because of short range communication, which is a must for mobile payments. And finally IT industry is interested to integrate many services, such as payment, loyalty, identification, access control and so on, and NFC technology is a response to that demand.

2.1.3 Internet of things and physical browsing

Before get deeper into NFC technology we should get to know to concept of Internet of things. When we talking about the Internet of things we could assume that all objects have identifier, identifier gives information about the object and the user access and uses this information in easy and fast manner.

The Internet of things refers to uniquely identifiable objects and their virtual representations in an Internet-like structure. The term Internet of Things was first used by Kevin Ashton in 1999. The concept of the Internet of Things first became popular through the Auto-ID Center and related market analysts publications. Radio-frequency identification (RFID) is often seen as a prerequisite for the Internet of Things. If all objects and people in daily life were equipped with radio tags, they could be identified and inventoried by computers. However, unique identification of things may be achieved through other means such as barcodes or 2D-codes as well. One of important things in concept of Internet of things is simplicity of usage.

One way to provide natural interaction between the user and technology is the concept of physical browsing, i.e. to implement interaction methods by which the user can achieve her or his goals, such as acquiring information, transferring data, performing

financial transactions or initiating actions, just by pointing or touching. The concept of using intuitive means of touching or pointing as a way of interfacing between a human and digital resources has been present in research at least since the 1990s. In the physical browsing concept personal device used for interacting with the environment has capabilities for user interfacing, such as display, sound and keypad; read and write memory; processing power and communication capabilities other than those needed for physical browsing. An example of such a personal device is a mobile phone with the ability to read (and write) NFC tags with an incorporated NFC reader.

As we see Internet of things and physical browsing concepts are closely related with NFC technology.

2.1.4 NFC use cases

We can envisage following application categories for physical browsing, and thus, for NFC technology.

One of categories is information retrieval. The information retrieval may be location and situation independent, such as getting product information by reading e.g. a web address from a NFC in a Smart Poster or it can be location or item specific, such as finding the time tables or next arrivals information on a specific bus stop.

Another category is value transactions. This application category consists of two main types: ticketing and payment. Some researchers see potential for great changes in the payment infrastructure in developing countries, where the users can skip the debit and credit card phase as well as bank transfers and substitute cash directly with NFC and mobile phone based money transactions. Ticketing refers to applications such as mass transit, sport events, concerts and movies, where the value is usually prepaid and usable only for limited and a priori known services. From user perspective this usually means a shift from a dedicated smart card, such as bus card, to emulation of that card inside the user's mobile phone. Payments are more general and can be seen substituting both cash based and debit card based (small) payments in cafeterias, kiosks, and shops. Mobile payment and ticketing is seen as the potential "killer application" for NFC technology. Prototype of parking system could be given as an example of value transaction service

(Figure 1). Smart Parking project was done at autumn 2007 by Top Tunniste, VTT and city of Oulu. I took part in implementation of mobile application for end users and parking controllers and also developed server side and web pages.

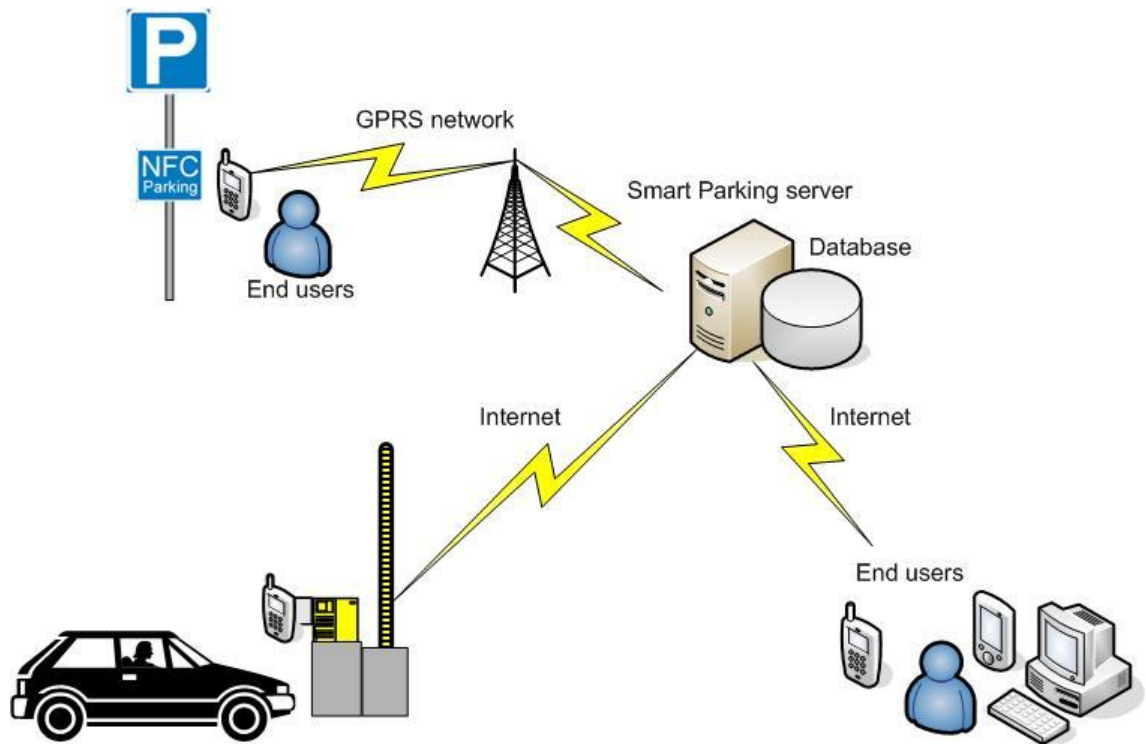


Figure 1. Smart Parking System

Third category is initiating of action. Many digital and physical operations can be initiated and even controlled using physical browsing paradigm. Simple examples include initiating a phone call by touching a business card or a person's photo – both equipped with a NFC tag, or ordering a taxi to certain place by touching a special service tag. The uploading of photos from a camera to a home server can be initiated easily by touching the home PC with the camera, if both are NFC equipped. This is also an example of peer-to-peer application, where no commercial partner is, at least directly, involved.

Fourth category is creation of social networks. We can envisage using NFC technology for establishing a digital link between persons who have established similar link in real world, for example people meeting in a business meeting or in a disco. Generation of digital links in the real world (by touching) is something between traditional means of networking (face to face) and web based networking, such as Facebook or LinkedIn.

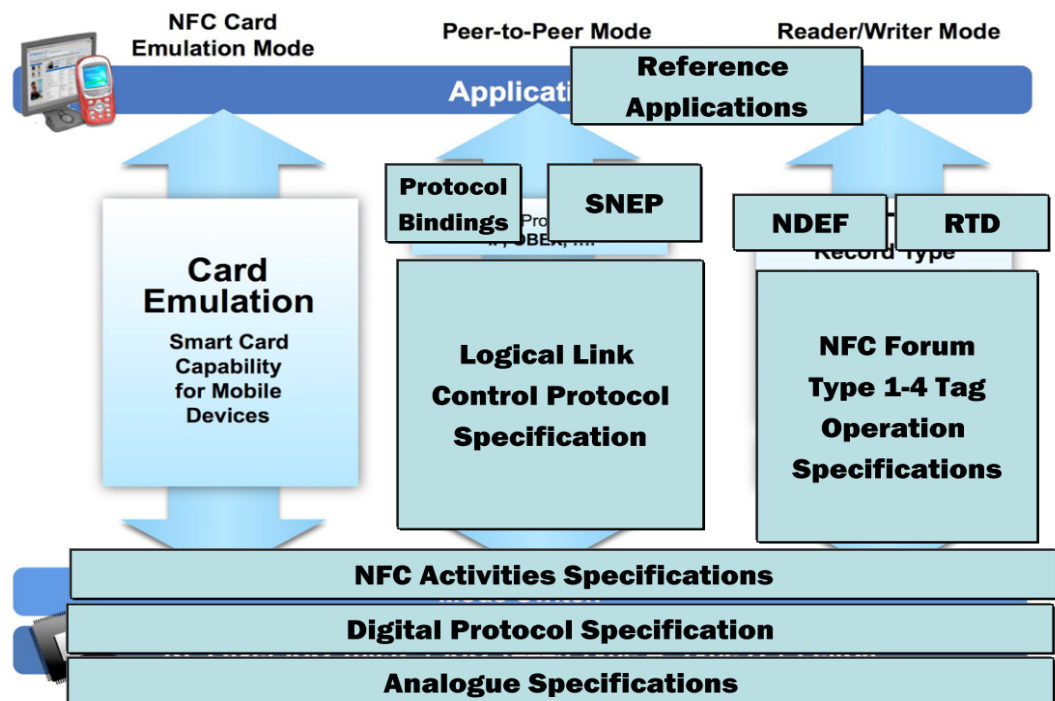
Fifth category is field force solution. Reporting with mobile phone using NFC tags instead of hand written reports is reducing time of creating reports for employees and

managers and controls field force staff actions. Those solutions could be used in waste management, shop management, security and manufacturing. One of first of such projects Rosk'n'roll was implemented in Finland by Nokia and ToP Tunniste in 2004.

All of the mentioned application categories rely on the intuitiveness of NFC based user interface paradigm. The use must be natural, easy and effective, when compared to other means for achieving the user's goals. For example, payment should be easier and faster than payment with a debit card or cash. Furthermore, the user must be able to feel secured that his money and privacy are not jeopardized.

2.2 NFC architecture

In June 2006, the NFC Forum introduced standardized technology architecture, initial specifications and tag formats for NFC-compliant devices. These include Data Exchange Format (NDEF), and three initial Record Type Definition (RTD) specifications for smart poster, text and Internet resource reading applications (Picture 1).



PICTURE 1. NFC architecture (NFC Forum <http://www.nfc-forum.org>)

In addition, the NFC Forum announced the initial set of four tag formats that all NFC Forum-compliant devices must support. These are based on ISO 14443 Types A and B (the international standards for contactless smartcards) and FeliCa (conformant with the ISO 18092, passive communication mode standard). Already more than one billion tags of this kind have been deployed globally, albeit for non-NFC applications like mass transit and access control.

The NFC technology architecture is based on three main modes of operation: peer-to-peer, read-write and card-emulation. NFC devices use the peer-to-peer mode for data transfer between devices, such as passing contact information or an electronic business card from one device to another—for example, between an NFC-enabled phone and an NFC-enabled PC. They can also use this mode for initially pairing, or linking, two devices that can then communicate via Bluetooth or another protocol. With read-write mode, an NFC-enabled device can access data from an RFID-enabled object, such as a "smart poster" with an embedded RFID tag that allows users to download a URL for a movie trailer. In the card-emulation mode, an NFC-enabled device emulates a contactless payment card and can be used to purchase goods and services.

Each of these modes requires that NFC devices use a common data format for communications. The first of these format specifications—the NFC data-exchange format (NDEF) and NFC record-type definition (RTD). The record data definition lets NFC devices talk to each other, with the data-exchange format establishing the grammar that they will use.

These two specifications standardize how NFC devices operate in the read-write mode. Three other specifications, describe how NFC devices in read-write mode will conduct specific transactions. There is a text RTD for exchanging records that include plain text; a Smart Poster RTD for using an NFC device to pull text, audio or other content from a Smart Poster; and a uniform resource identifier (URI) RTD for exchanging records that refer to Internet resources.

2.2.1 Three modes of NFC

First mode is read-write mode (Figure 2). Initiation of service requires two devices to communicate using NFC; one device is an NFC reader/writer and the other a passive NFC tag. At a hardware level, there is at least one chip, called the NFC radio. In order to run secure applications such as payment, transport ticketing, or access to buildings, there is need for a second chip, called the Secure Element (SE). Functionality of Secure element could be also implemented in USIM.

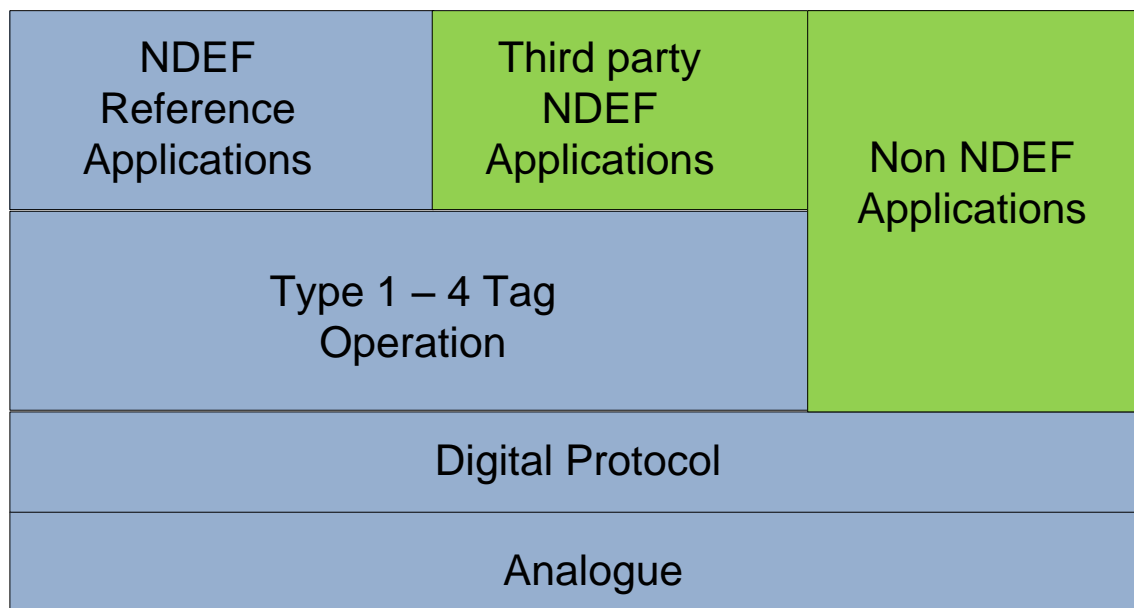


FIGURE 2. Reader/Writer mode

The NFC radio can interact with Tags, Readers, or other NFC Peers when they are in close proximity to one another (typically less than 10cm). This NFC radio is connected to a Host controller, which can be the Baseband or Application Processor on a phone or the core CPU in a PC.

A Secure Element (SE) is separate chip or USIM which contains a secure processor, tamperproof storage and execution memory. This processor is different from the Host processor or PC processor. Its sole purpose is enabling secure transactions. The Secure Element contains applications which rely on secure keys running inside the secure processor. The applications which run on the secure element typically run on a JavaCard applet.

Peer-to-peer mode enables two NFC enabled mobile devices to establish a bidirectional connection and exchange information such as a contact record, a text message, or any other kind of data (Figure 3). In this mode, NFC is comparable to other short-range communication technologies such as Bluetooth, Wibree and IrDA, although the physical data transfer mechanism is different. In this respect, NFC can be seen as a rival of these technologies, even though it can also complement them. NFC can open a connection between two devices that are brought close to each other, and the actual communication will then occur by Bluetooth or WLAN. This mode has two standardized options: NFCIP-1 and LLCP.

Ref Apps	Applications	
Simple NDEF Exchange Protocol	NFC Forum Registered Protocols	Other Protocols
	Protocol Bindings	
Logical Link Control Protocol		
Digital Protocol		
Analogue		

FIGURE 3. Peer-to-peer mode

NFCIP-1 is standardized in ISO/IEC 18092, ECMA 340, and ETSI TS 102 190. This standard defines two communication modes as active and passive. It also defines the RF field, RF communication signal interface and general protocol flow, initialization conditions for the supported data rates of 106, 212, and 424 kbps in detail. Moreover, it defines transport protocol including protocol activation, data exchange protocol with frame architecture and error detecting code calculation (CRC for both communication mode at each data rate), and protocol deactivation methods. The carrier frequency of the RF field is 13.56 MHz. The active communication mode, both the initiator and target use their own RF field to enable communication. The initiator starts the NFCIP-1 communication whereas the target responds to an initiator command in the active communi-

cation mode using self-generated modulation of the self-generated RF field. The target is powered by inductive coupling and is able to send and receive data.

In the passive communication mode, the initiator generates the RF field and starts the communication. The target responds to an initiator command in the passive communication mode using a load modulation scheme. The communication scheme over the RF interface in active and passive communication modes includes modulation schemes, transfer speed and bit coding. Additionally it includes start of communication, end of communication, bit and byte representation, framing and error detection, single device detection, protocol and parameter selection, data exchange and de-selection of NFCIP-1 devices. All NFCIP-1 devices have communication capability on 106, 212, or 424 kbps and may switch to another transfer speed or stay on the same transfer speed. The transfer speed of the initiator to target and the transfer speed of the target to the initiator do not need to be kept the same during a transaction. The change of transfer speed during a transaction session may be performed by a parameter change procedure. The mode (active or passive) cannot be changed within one transaction session. The transaction is started by device initialization and terminated by device de-selection (or equivalent).

NFCIP-2 (specified in ECMA-352) defines how to automatically select the correct operation mode when starting communications. In peer-to-peer mode, the participant that starts the communication is called the initiator and the other participant the target. The peer-to-peer mode is divided into two variants: active mode and passive mode. In active mode, both participants generate their own carrier while transmitting data. In passive mode, only the initiator generates a carrier during communications, and the target device uses load modulation when communicating back to the initiator, in a way similar to passive RFID tag behaviour. This makes it possible to save power in the target device, which is a useful feature if the target device has a very restricted energy source, such as a small battery. Fundamentally it is possible to make a target device – such as a battery assisted (semi passive) sensor readable over NFC – last for several years, even if operated from a small lithium coin-cell battery. Battery less (passive) sensors that are powered by the RF field of an active NFC device is also feasible.

After some intensive debate at the NFC-Forum, the following protocol has finally been accepted for becoming the P2P supporting standard. This protocol, named LLCP. The LLCP defines an OSI data link protocol to support peer-to-peer communication between

two NFC enabled devices. LLCP is essential for any NFC application that involves a bi-directional communication. It enhances the basic functionalities provided by NFCIP-1 protocol as well. LLCP provides five important services: connectionless transport; connection oriented transport; link activation, supervision and deactivation; asynchronous balanced communication; and protocol multiplexing.

In card emulation mode, the NFC enabled device acts as a smart card (Figure 4). Either an NFC device emulates an ISO 14443 smart card or a smart card chip integrated in a mobile phone is connected to the antenna of the NFC module. More detailed description of card emulation mode is described in Chapter 3.1.2

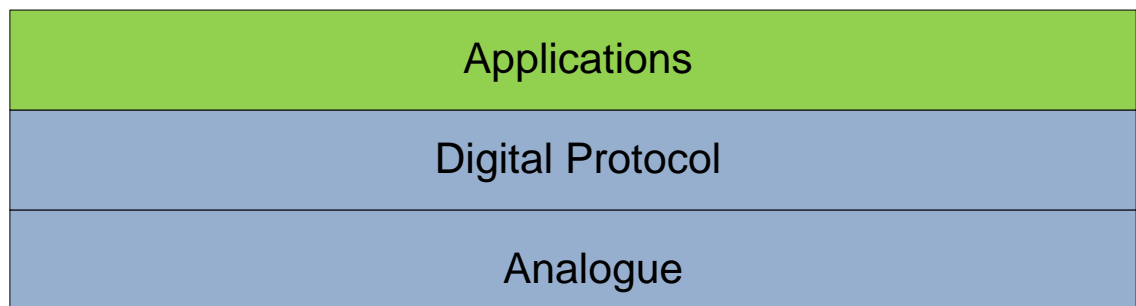


FIGURE 4. Card emulation mode Architecture

2.2.2 NFC Data Exchange Format

The NFC Data Exchange Format (NDEF) specification defines a message encapsulation format to exchange information, e.g. between an NFC Forum Device and another NFC Forum Device or an NFC Forum Tag.

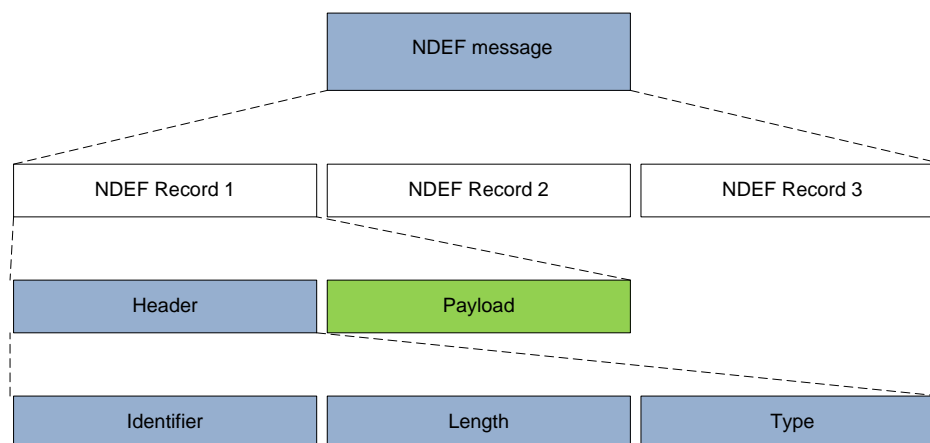


FIGURE 5. NDEF message structure

NDEF is a lightweight, binary message format that can be used to encapsulate one or more application-defined payloads of arbitrary type and size into a single message. Each payload is described by a type, a length, and an optional identifier.

Type identifiers may be URIs, MIME media types, or NFC-specific types. This latter format permits compact identification of well-known types commonly used in NFC Forum applications, or self-allocation of a name space for organizations that wish to use it for their own NFC-specific purposes. An NDEF message is composed of one or more NDEF records. There can be multiple records in a NDEF message. Basically NDEF message is array of NDEF records. Amount of records we can encapsulate in a NDEF message depends on our application and the tag type. NDEF is strictly a message format, which provides no concept of a connection or of a logical circuit, nor does it address head-of-line problems.

2.2.3 Record Type Definition

Record Type Definition is an NFC-specific record type and type name which may be carried in an NDEF record. The record type string field of an NDEF record contains the name of the record type (called “record type name”). Record type names are used by NDEF applications to identify the semantics and structure of the record content.

Record type names may be specified in several formats, called Type Name Formats, as signified by the TNF field of the NDEF record header. Record type names may be MIME media types, absolute URIs, NFC Forum external type names, or may be well-known NFC type names (RTD’s, the subject of this specification). Each record type definition is identified by its record type name.

The NFC Forum Well-known Type is a dense format designed for tags and creating primitives for certain common types. It is meant to be used in case there is no equivalent URI or MIME type available, or when message size limitations require a very short name. The Namespace Specific String of the NFC Well Known Type URN is prefixed with “wkt:”. For example, the Well Known Type “urn:nfc:wkt:a” would be encoded as

“a”. The Well Known Type “urn:nfc:wkt:Very-complicated-type” would be encoded as “Very-complicated-type”. There are two type of WKT, one is NFC Forum Global type, starts with an upper-case letter, e.g. “U” and NFC Forum Local type, start with a lower-case character in character set or with a number in character set which can be found in RTD specification e.g. “2”, “trip”.

The External Type Name is meant for organizations that wish to self-allocate a name space to be used for their own purposes. The External Type Name must be formed by taking the domain name of the issuing organization, adding a colon, and then adding the type name as managed by the organization. A canonical version of the External Type Name would look like: “urn:nfc:ext:example.com:f”. NFC Forum defines various record types for specific cases; smart posters, URIs, digital signature, and text.

The URI Service RTD is an NFC RTD describing a record to be used with the NDEF to retrieve a URI stored in a NFC-compliant tag or to transport a URI from one NFC device to another. The URI (either a URN or URL) also provides a way to store URIs inside other NFC elements, such as a Smart Poster. The Well Known Type for an URI record is “U”. The structure of an URI record has URI Identifier Code, the first byte of the record data describes the protocol field of an URI and URI field, which can be a URL or URN.

The Text Record Type Description defines an NFC Forum Well Known Type for plain text data. It may be used as free form text descriptions of other objects on an RFID tag. The “Text” record contains freeform plain text. The Text record may appear as a sole record in an NDEF message, but in this case the behavior is undefined and left to the application to handle. Typically, the Text record should be used in conjunction with other records to provide explanatory text. The NFC Forum Well Known Type for the Text record is “T”. In text record types, the text can be encoded in either UTF-8 or UTF-16, which is defined by the status byte in the text record. The text record is composed typically of a NDEF record header, a payload and the actual body text in UTF format.

The Smart Poster Record Type Definition defines an NFC Forum Well-known Type on how to put URLs, SMSs, or phone numbers on an NFC Forum Tag or how to transport them between devices. More detailed information is described in different chapter.

The Signature RTD specifies the format used when signing single or multiple NDEF records. Digital signing of NDEF data is a trustworthy method for providing information about the origin of NDEF data in an NFC Forum Tag and NFC Forum Device. The Signature record contains a digital signature related to one or more records within an NDEF message. The signature can be used to verify the integrity and authenticity of the content, i.e., the data records that have been signed. The NFC Forum Well Known Type for the Signature record is "Sig". The contents of the payload of a Signature record consist of the following fields: Version, Signature, and Certificate Chain.

2.2.4 NFC Forum tag types

Four tag types have been defined by NFC Forum, and are given designations between 1 and 4. Each tag type has a different format and capacity. NFC tag type formats are based on either ISO/IEC 14443 Type A, ISO/IEC 14443 Type B, or Sony FeliCa.

Type 1 is based on ISO 14443 A standard (available from Innovision Research & Technology (Topaz™)). Memory availability is up to 1 kB which is just enough to store a website URL or similar small amount of data. The memory size can be expanded up to 2 kB. The communication speed of this NFC tag is 106 kbps. As a result of its simplicity, this tag type is cost effective and can be used in many NFC applications.

An NFC Forum Type 1 Tag can be classified to exist in one of the following states: INITIALIZED, READ/WRITE, and READ-ONLY. READ/WRITE state means that tag contains a valid NDEF message. It is available for read and re-write access. READ-ONLY state means that tag contains a valid NDEF message and be available for read-only access. It cannot be deleted or overwritten with a new NDEF message. Tag in INITIALIZED state when not in the READ/WRITE or READ ONLY states.

Type 2 is also based on ISO 14443 A standard (MIFARE Ultralight, available by NXP). The memory structure is divided in blocks containing 4 bytes each. Each block is numbered from 0 to 15 for static memory structure or from 0 to k for dynamic memory structure. The blocks are grouped in sectors. First four blocks contain tag UID and static lock information. The rest of memory is used for data reading and writing. Type 2 tags have similar states as Type 1.

Type 3 is based on FeliCa standard (available by Sony). It operates at a higher data rate (212kbit/s) and has a larger memory. The basic unit of information used in memory management is called a block. Each block has a fixed size of 16 bytes. The number of memory blocks available depends on the chip hardware. A Type 3 Tag has only one state called “Mode 0”. In this state, the Polling Command, Check Command, and Update Command can be received. None of these Commands change the state of the Type 3 Tag. Memory blocks are not addressed directly but relative to the Service they belong to. Services are similar to files in a file system. Type 2 tag is suitable for more complex applications.

Type 4 is fully compatible with ISO 14443A/B and is available from a number of manufacturers, including NXP (typical product example is MIFARE DESFire). The Type 4 Tag Platform contains at least the NDEF Tag Application. The NDEF Tag Application contains the NDEF messages on a Type 4 Tag Platform that provides a file system composed of at least two EF files: the Capability Container file and the NDEF file. It offers large memory-addressing capability with read speeds of between 106kbit/s and 424kbit/s – making it suitable for multiple applications.

2.3 NFC devices and NFC development

Finland is leader in NFC innovations. One of the first commercial NFC devices presented to the world become Nokia 6210 and Nokia 5140 with NFC shell. ToP Tunniste was one of the companies who developed first pilot NFC services with cooperation with Nokia. Before NFC Forum released standards for NFC, Nokia already provided tools and devices for NFC developers. Since then many new NFC devices were released along with different development environments and NFC specifications.

2.3.1 NFC Development

There are a lot of different platforms and SDK's which support NFC development. Developing NFC-based applications for Java handsets is based on the Mobile Information Device Profile together with Contactless API (JSR 257) for contactless I/O, and SATSA (JSR 177) for secure element capabilities. Phones which support JSR 257: (Nokia C7, Nokia 701, Nokia 6212, Nokia 6131).

The Android NFC API was introduced with API Level 9 (Android 2.3). The `android.nfc` package provides access to Near Field Communication (NFC) functionality, allowing applications to read NDEF message in NFC tags. The API follows the NFC Forum concepts such as NDEF Messages, Records and Tags.

Libnfc is a mature, cross-platform, open-source NFC library for desktop C++ development.

The Connectivity API provides a set of APIs for QT C++ development. The current version of the Connectivity API includes support for Bluetooth and Near Field Communication (NFC) technologies. The NFC API provides APIs for interacting with NFC Forum Tags and NFC Forum Devices, including target detection and loss, registering NDEF message handlers, reading and writing NDEF messages to NFC Forum Tags, send tag specific commands. The API also provides client and server LLCP sockets.

Windows Phone 8 SDK provides Proximity API for NFC development. Proximity refers to a set of classes in the Windows Runtime that support connections between devices that are within close range of each other. Windows Phone 8 supports Proximity communication using Near Field Communication (NFC). Since Proximity API gives only high level access to the NFC protocol and Windows Phone adds some protection on top of that, interaction with NFC tags is limited. Only tags with NDEF are supported, it is not possible to write lock tags and format tags.

2.3.2 NFC devices overview

There are a lot of different NFC devices exists today. NFC chip is integrated in many home devices like TV's, speakers, refrigerators. But of course most of NFC devices are desktop NFC readers, NFC mobile phones and NFC tags. NFC works in a very intuitive way. Two NFC devices immediately start their communication as they are touched. The touching action is taken as the triggering condition for NFC communication. Hence, the user does not need to interact with the mobile device anymore but just touches one appropriate NFC device which may be an NFC tag, an NFC reader, or another mobile phone. We can classify the NFC devices in the communication based on two parameters. The first parameter is the energy supply which results in active and passive de-

vices. The second one is initiating the communication and leads to initiator and target devices.

Active and passive device definitions are important to understand the NFC communication. An active device is one that is powered by some power source – such as a battery – so that it generates its own electromagnetic field. On the other hand, a passive device is one that does not have any integrated power source. It is a rule of nature that each activity requires energy; hence even a passive device requires some power to perform as programmed previously. In NFC communication the energy is supplied by the other (active) party for the passive device. To summarize, an active device powers the passive device by creating the electromagnetic field.

NFC always occurs between two parties, so that one party is called the initiator, and the other party is called the target. The initiator is the one that initiates the communication; the target responds to the request that is made by the initiator. An initiator obviously always needs to be an active device, because it requires a power source to initiate the communication. The target, on the other hand, may be either an active or a passive device. If the target is an active device, then it uses its own power source to respond; if it is a passive device, it uses the energy created by the electromagnetic field which is generated by the initiator that is an active device.

2.3.3 NFC mobile device architecture

A Mobile NFC device provides a combination of NFC services with mobile telephony services. With this definition, a mobile phone employing a UICC as the Secure Element and an NFC controller can provide an ideal platform for NFC service applications.

Within this context, the UICC hosts and manages the NFC service files and applications, while the NFC controller manages the NFC interface (i.e. to the contactless reader). Figure 6 below depicts the NFC mobile device as three main components: the application processor, the NFC chip and the UICC card. The UICC contains a hierarchy of Security Domains in accordance with the Global Platform standard: The Issuer security domain created for the issuer of the secure element (in this case the MNO) and the

Secondary Security Domains created on demand for the different Service providers that have installed their applications in the UICC.

The UICC (USIM), as used by MNOs, is based on ETSI Smart Card Platform standards which aim to comply with Global Platform standards. The Global Platform model defines a hierarchical structure of privileges, addressing the issue of extending UICC resources to third parties.

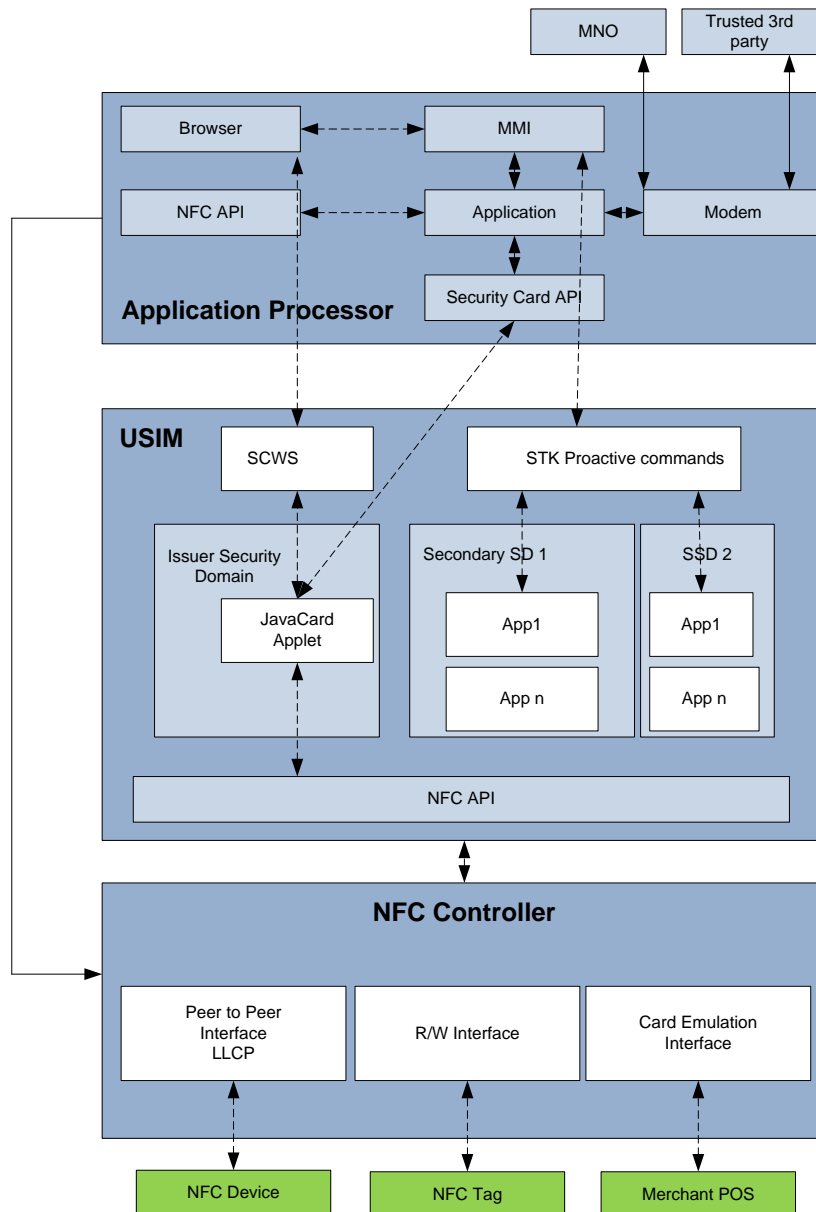


FIGURE 6. NFC device architecture

NFC chips (controllers) are generally manufactured by semiconductor companies. And other hardware companies built products using these NFC controllers. The hardware

manufacturer also develops the software API which is used by the operating system of the final product for accessing the hardware resources of the NFC controller. In the mobile device NFC controller is responsible for switching and routing between different NFC modes. When the UICC needs to start a transaction it has to be switched into the correct mode.

2.4 NFC security aspects

Although the communication range of NFC is limited to a few centimetres, NFC alone does not ensure secure communications. In 2006, Ernst Haselsteiner and Klemens Breitfuß described different possible types of attacks, and detail how to leverage NFC's resistance to Man-in-the-middle attacks to establish a specific key. Unfortunately, as this technique is not part of the ISO standard, NFC offers no protection against eavesdropping and can be vulnerable to data modifications. Applications may use higher-layer cryptographic protocols (e.g., SSL) to establish a secure channel. Ensuring security for NFC data will require the cooperation of multiple parties: device providers, who will need to safeguard NFC-enabled phones with strong cryptography and authentication protocols; customers, who will need to protect their personal devices and data with passwords, keypad locks, and anti-virus software; and application providers and transaction parties, who will need to use anti-virus and other security solutions to prevent spyware and malware from infecting systems.

Because NFC is a wireless communication interface it is obvious that eavesdropping is an important issue. When two devices communicate via NFC, they use RF waves to talk to each other an attacker can of course use an antenna to also receive the transmitted signals. When a device is sending data in active mode, eavesdropping can be done up to a distance of about 10 m, whereas when the sending device is in passive mode, this distance is significantly reduced to about 1 m.

Instead of just listening an attacker can also try to modify the data which is transmitted via the NFC interface. In the simplest case the attacker just wants to disturb the communication such that the receiver is not able to understand the data sent by the other device.

Data corruption can be achieved by transmitting valid frequencies of the data spectrum at a correct time. The correct time can be calculated if the attacker has a good understanding of the used modulation scheme and coding. This attack is not too complicated, but it does not allow the attacker to manipulate the actual data. It is basically a Denial of Service attack.

In data modification the attacker wants the receiving device to actually receive some valid, but manipulated data. This is very different from just data corruption. The feasibility of this attack highly depends on the applied strength of the amplitude modulation. This is because the decoding of the signal is different for 100% and 10% modulation. It is much more difficult to modify data in such a way that it appears to be valid to users. To modify transmitted data, an intruder has to deal with the single bits of the RF signal.

Data Insertion means that the attacker inserts messages into the data exchange between two devices. But this is only possible, in case the answering device needs a very long time to answer. The attacker could then send his data earlier than the valid receiver. The insertion will be successful, only, if the inserted data can be transmitted, before the original device starts with the answer. If both data streams overlap, the data will be corrupted.

Losing the NFC RFID card or the mobile phone will open access to any finder and act as a single-factor authenticating entity. Mobile phones protected by a PIN code acts as a single authenticating factor. A way to defeat the lost-property threat requires an extended security concept that includes more than one physically independent authentication factor.

In the classical Man-in-the-Middle Attack, two parties which want to talk to each other are tricked into a three party conversation by an attacker. Such a setup is the classical threat in unauthenticated key agreement protocols like Diffie-Hellmann protocol.

There are following common attacks against NFC tags: tag cloning and tag impersonation, tag content changes, tag replacement and tag hiding. From the RFID technology point of view, the most challenging security threats in commercial RFID applications are tag cloning and tag impersonation. The research community addresses these threats primarily by trying to make tag cloning harder by using cryptographic tag authentication.

tion protocols. The fundamental difficulties of this research revolve around the trade-offs between tag cost, level of security and performance in terms of reading speed and distance. It is currently challenging to protect a passive RFID tag from cloning.

Tag cloning has the same impact as relay attacks with the big advantage to the attacker that the RFID tag can be reused without the need to use the victim for each individual attack. Note that a relay attack is performed during an NFC communication, whereas cloning is performed before the communication starts. Hence, it technically seems easier to perform tag cloning.

Sticking a malicious tag on top of the original tag or replacing the original tag with a malicious tag is enough to let the system work as the attacker desires. In the case of sticking on a new tag, it is possible to disable the old tag. Another method to attack passive tags is to break the write protection of the tag and overwrite it with malicious data.

3 NFC Tag management and audit

This chapter describes tag management system and technologies it using, models and concepts, use case scenarios, business aspects, security and other issues.

3.1 Smart Poster and Tag emulation

Tag management system handles different NFC tags which are located in the streets, premises and homes. Those NFC tags could be passive or active ones or they could be emulated by NFC reader or mobile phone. My prediction that most of tags will be Smart Poster tags. In this subchapter I will describe more specifically Smart Poster and tag emulation.

3.1.1 Smart Poster

Smart Poster Technical Concept has been developed by the NFC Forum. It defines how phone number, SMS or URL can be stored in a NFC tag and transport them between devices. This way information and actions can be attached in a Smart Poster. Basically this allows transforming any physical object to a smart object by attaching an NFC tag in it. This means that the object can store additional information about itself.

The Smart Poster makes possible to initiate a phone call, send SMS or go to URL by reading a NFC tag with a NFC phone. The Smart Poster can contain actions, which trigger an application in the NFC phone. The technical solution makes also possible to edit or save the information (phone number, SMS, URL) read from the NFC tag.

The Smart Poster concept is built around URIs, which have become the standard for referencing information around the Internet. URIs are very powerful, and they can represent anything from unique identifiers to EPC codes to web addresses to SMS messages to phone calls and beyond. The Smart Poster Record defines a superstructure that associates a URI with various types of metadata. There is only one URI record per Smart Poster record. This is also the only mandatory record within a Smart Poster.

The content of a Smart Poster payload is an NDEF message. The contents of this message consist of several NDEF records.

NFC Smart Poster should have touch point which is clearly indicated, and that there is an adequate description of what users will receive when they interact with the poster.

There can be more than one touch point per object, providing access to different digital services and enhancing the functionality of the NFC Smart Poster. It's important that the instructions make it clear which touch point engages which service.

Programming and content management for NFC Smart Poster can be as simple or more complex. The content accessed via touch point on an NFC Smart Poster can be minimal and static, such as a website address or a phone number for later use. Or, the content can be dynamic, so that the end user interacts with the content provider, once a connection is made. In this case, additional content beyond what is on the NFC tag is provided over the air to the end-user's NFC device, such as an NFC mobile handset.

At a higher level of complexity, NFC Smart Posters the content sent to a user can be changed in a backend system as needed (new items on sale by a retailer every week, for example), again without having to rewrite the NFC tags.

Using a backend management system with an NFC Smart Poster can increase the amount of available content far beyond the capacity of a single tag, such as by including pictures and movies, or adding levels of interactivity.

3.1.2 Tag Emulation with NFC reader and mobile phone

In card emulation mode, the NFC enabled mobile phone acts as a smart card. Either an NFC enabled mobile phone emulates an ISO 14443 smart card or a smart card chip integrated in a mobile phone is connected to the antenna of the NFC module. As the user touches mobile phone to an NFC reader, the NFC reader initiates the communication. NFC devices that are operating in card emulation mode use similar digital protocol and analog techniques as smart cards and they are completely compatible with the smart card standards. Card emulation mode includes proprietary contactless card applications such as payment, ticketing and access control. These applications are based on ISO/IEC 14443 Type A, Type B and FeliCa communication interfaces.

There are two standard protocols adopted by the ETSI/3GPP and aiming at implementing the NFC card emulation mode in Mobile Handset terminals also provided with a SIM or UICC card. The first one, known as SWP (Single Wire protocol) is a communication interface between the SIM/ UICC and a contact-less frontend (CLF) in the terminal. This interface allows the card emulation mode independent of the power state of the terminal as well as the reader mode when the terminal is battery powered. It is to ensure interoperability between a UICC and the CLF in the terminal independently of the respective manufacturer, card issuer or operator.

The second one, known as Host Controller Interface specifies a logical interface that enables contact-less applications hosted on the UICC. It especially covers the configuration where the one host is embedded in the UICC which is connected to the host controller embedded in the CLF.

The Single Wire Protocol is a digital full duplex protocol based on a single active wire. The data rate is scalable from 212 Kb/s until 1.6 Mb/s for short distance less than 10 cm. On the top of the physical layer, a bit oriented MAC layer based on a tailored HDLC (ISO/IEC 13239) is implemented. The SWP protocol is able to carry short packets (payload less than 30 bytes) to several nodes and allow the routing of messages to different entities (e.g. The RF front end, the settings, the host terminal...) in a proactive manner.

The Host Controller Interface is a logical interface, allowing an NFC front end to communicate directly with an application processor and multiple secure elements in various electronic devices such as mobile phones and PC peripherals, enabling faster integration of NFC functionality. The HCI defines so the interface between logical entities that operates one or more service(s), also called “hosts”. According to the ETSI terminology, a network of two or more hosts is called a “host network”, and one of the “hosts” that is also responsible for managing a host network is called the “host controller”.

3.2 Tag management

NFC ecosystem develops rapidly, millions and millions of tags already in use and it's clear that there is a need to manage all those tags in different ways. It's possible to have

tags with static link or tags could be dynamically associated with a service, campaign or product. Tag management system offers way to create and manage functionality and content, which can be reached by reading NFC tags or QR codes (Figure 7). There are many tag management systems appears at the market recently, e.g. Microsoft Tag.

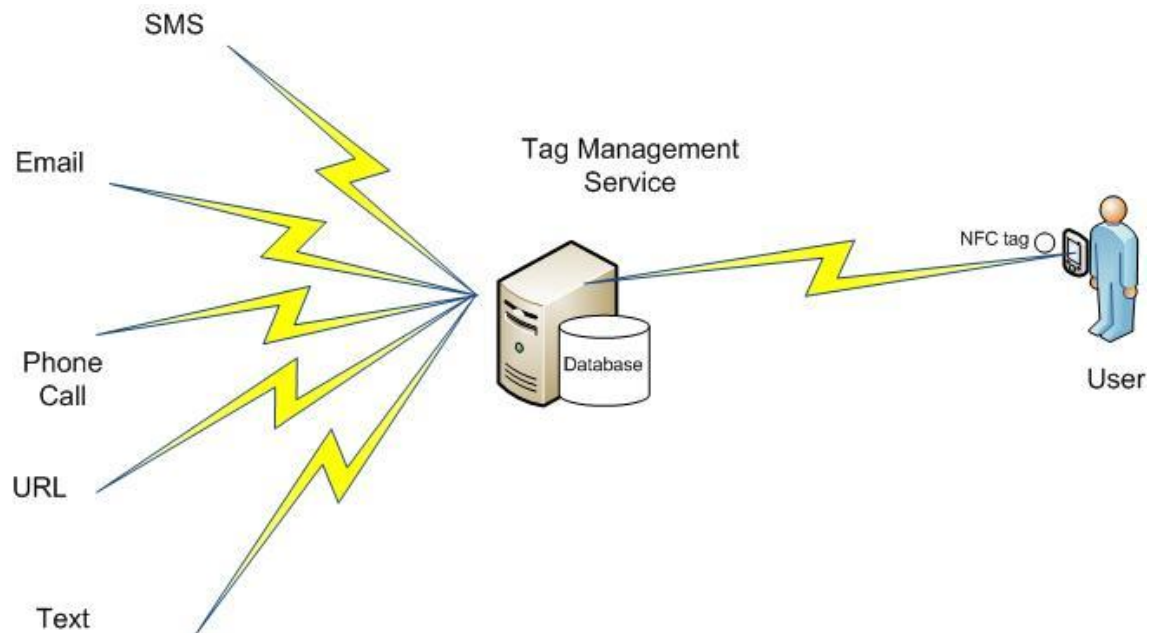


FIGURE 7. Tag Management System

Tag management can manage different types of tags from UHF EPC and barcode to NFC tags. Important part of tag management is ability to track tags tapping in real-time, and access reporting information. It allows to measure how often users engage with tags, where they are, and what devices they're using.

Another useful feature of tag management system is information about physical location of tags. So then tags are encoded and delivered, it's crucial to know where they are located and be able to change its location, if tag physically moved to another place.

There are following use case scenarios for tag management system. It could be used for merchandising, shopping campaigns, infotainment services like bus schedules, museums and tourist attraction information, product description at different venues. It could be integrated with location based services and asset management systems.

3.2.1 Models and Concepts

As we described in previous chapters, data on the NFC tag is NDEF message, which contains RTD records. Most commonly used type of RTD record is Smart Poster and URI. Smart Poster record is NFC Forum well known type, it has two or more RTD records, which are usually Title record and URI record. There are two possible ways of managing tag content: static links and dynamic links. Let's discuss pros and cons of each model.

Static links have direct URI, phone number call action, SMS or email. Once tag is encoded and placed at poster or other location, owner of tag can't anymore manage tag content. Of course if tag is not write-protected it possible to go to location and manually change NDEF message. But in case of huge amount of tags, located at different places or even cities and countries, it will be almost impossible. Hence static links are suitable only for time limited campaigns, where no need to further manage those tags.

Dynamic links contain URI, which is pointed to tag management system. Content can be provided dynamically through two approaches. In one, the NFC device and a pre-loaded application drive the selection of content, through stored information or preferences. When the connection is made, the information stored on the NFC device results in the delivery of data tailored to the user. For example, a clothing shop's application would call for information on male clothing for male users, or multi-lingual messages could be provided according to the preferences set up on the NFC device.

The second way to provide and manage dynamic content is by using a backend system, which can be very convenient in practice. In this case NFC tags have certain ID, which identifies tag in that URI, it could be unique tag UID created by manufacturer of tag or it could be some another ID created by owner of tag. Group of tags could have same ID encoded in URI, so they treated as one in tag management system. One disadvantage of same ID in many tags is that it's not possible to distinguish them and certainly to know where and what tag was tapped. Dynamic links has some advantages over static links, it's possible to manage tags and change where content depending of time of the day, day of the week and other rules defined by owner of tags.

When we talking about tag management, we should describe two concepts. First concept is one tag one service. It means that tag belongs to one system and one owner. Tag content is defined by that system and managed by tag owner. Second concept is one tag many services. In this case tag could share between two or more systems and managed by many owners. For example, tag at bus station, could be used by users to get bus schedules and at the same time, it could be used by cleaning company to report work done for this location. Cleaning company personnel should have mobile application which reads tag URI, retrieves ID and connects to server to make report. This concept, one tag many services is quite important by my opinion and tag management system which will provide this functionality will have benefits over others tag management systems.

Let's discuss different models of accessing information and services. Access to content and services is provided by tags or by RFID readers emulating NFC tags. What types of identifiers could be used in tag management system? First type is quite familiar to all of us, it is barcode. A barcode is an optical machine-readable representation of data relating to the object to which it is attached. Originally barcodes systematically represented data by varying the widths and spacings of parallel lines, and may be referred to as linear or one-dimensional (1D). They widely used to mark products at shop.

Second type of identifiers is also barcode, but 2D barcode, and most common type of 2D barcode is QR code. QR code (abbreviated from Quick Response Code) is the trademark for a type of matrix barcode (or two-dimensional bar code) first designed for the automotive industry in Japan. Originally designed for industrial uses, QR codes have become common in consumer advertising. Smartphone users can install an app with a QR-code scanner that can read a displayed code and convert it to a URL directing the smartphone's browser to the website of a company, store, or product associated with that code providing specific information.

Third type of identifiers is UHF tags with EPC code. The Electronic Product Code (EPC) is designed as a universal identifier that provides a unique identity for every physical object anywhere in the world, for all time. Its structure is defined in the EPCglobal Tag Data Standard. EPCs are not designed exclusively for use with RFID data carriers. They can indeed be constructed based on reading of optical data carriers, such as linear bar codes and two-dimensional bar codes, such as Data Matrix symbols. The

EPC was developed at MIT's Auto-ID Centre in 2000 and is a modern-day replacement for the Universal Product Code (UPC). A tag's embedded EPC number is unique to that tag. Wal-Mart, like other large retailers, had more pragmatic issues at hand when they established an RFID requirement for their suppliers. Under Wal-Mart's mandate, each supplier is required to identify their products not by bar codes and waybills, but through EPCs that are automatically broadcast by RFID tags as new products arrive at the retailer's warehouse, distribution centre, or store.

Fourth type of identifiers is NFC tags. Four NFC tag types have been defined by NFC Forum, and are designated as Type 1, Type 2, Type 3 and Type 4. Each tag type has a different format and capacity. NFC tag type formats are based on either ISO 14443 Type A, ISO 14443 Type B, or Sony FeliCa. The most widely known type of NFC tags is MIFARE Ultralight, they are used by transport system and as access cards. Special case of providing content and services is NFC card emulation. In card emulation mode, an NFC enabled device acts as a smart card. NFC device could be used in tag management system, to provide dynamical tag content.

Above we described different types of tags used in tag management system. As we can see all those technologies and designed for different use cases and scenarios. In my opinion combining QR code, NFC tag and EPC in one tag is most promising way for tag management. Some of those technologies competing each other, some are designed for totally different things. It's not clear will EPC code or barcode prevail or QR codes and NFC will be mostly used to identify objects. At ToP Tunniste we developed prototype tag which combines UHF and HF tag with QR code. For next few years I think, it will be smart and flexible solution.

3.2.2 Business aspects

Business aspects describe way of monetizing tag management service and integration it with commercial location based, field force and ticketing systems. Tag management system could be used as platform for advertisement and promotional campaigns. It could be used by companies to offer their services and products; it could be integrated in asset management system.

Another business opportunity is to offer tags to other companies to be used with their services and applications, this concept one tag many services was described in previous chapter. One example of such cooperation is usage of tags by security companies. Security companies provide guarding services, which held during night, by visiting locations by guards. It's important to know at what time guard has visited location. Hence NFC tags could be used to report to the server time and location visited by guard. So different field force personnel, which provides maintaining, cleaning, guarding and other services could use existing NFC tags infrastructure.

Tag management could be integrated with location based system, e.g. by tapping tag, user will see map on his device, with additional information like lunch offers if its lunch time or night clubs and restaurants ads during evening. Companies could offer coupons and discounts for nearby shops. Cinemas and shopping malls could place advertisement related to the tag location.

3.2.3 Security and other issues

Tag management system includes physical tags, software and web services. So all threats and attacks which could be done against software or service, are also threatening tag management system. But it also has some issues and threats which are unique for it.

Physical replacement of tag with another tag with harmful content is one of threads for users. NFC tag or QR code could be replaced with another tag, which contains, for example, link to fishing web site, to get user credential or other important information (e.g. credit card number), which potentially could be used for a fraud.

Vandalism or physical damaging of tags is one issues arising for owners of the tags. ToP Tunniste, Deutsche Bahn (DB) and Rhein-Main-Verkehrsverbund (RMV), offered way to protect tags using plastic cover (Picture 2, bottom right). This plastic cover protects from fire and physical damage and proved to be reliable protection.



PICTURE 2. NFC tag protection

(Frankfurt, 2012-12-27, picture is taken by Alexei Chugunov)

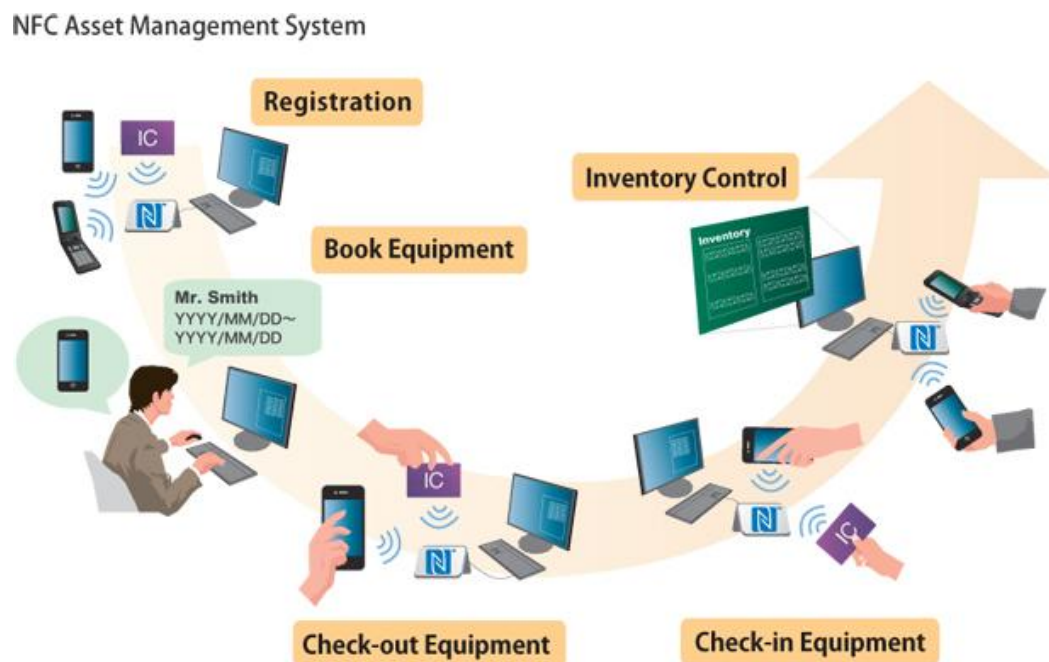
Changing content of tag is similar to replacement of the tag, only difference that physically it's the same tag which was before. Digital signature of NDEF data could be used to protect tag content from modification. NFC Forum has release NFC Signature RTD specification. Digital signing of NDEF data is a trustworthy method for providing information about the origin of NDEF data in an NFC Forum Tag and NFC Forum Device. It provides users with the possibility of verifying the authenticity and integrity of data within the NDEF message. Another way to protect data is write protection of tag, different tag manufactures provides way to protect data permanently or temporally with protection keys.

DoS (denial-of-service) attack could slow down or make system unavailable for a long period of time. Defending against Denial of Service attacks typically involves the use of a combination of attack detection, traffic classification and response tools, aiming to block traffic that they identify as illegitimate and allow traffic that they identify as legitimate.

Another threat is gaining unauthorized access to web service and database. It could be done using password cracking, which is done by repeatedly try guesses for the password or by using web servers or web tools vulnerabilities. After that hacker could modify, delete or add data into database or retrieve important information.

3.3 Asset management with NFC

Asset management, broadly defined, refers to any system that monitors and maintains things of value to an entity or group (Picture 3). It may apply to both tangible assets such as buildings and to intangible concepts such as intellectual property and goodwill. Asset management is a systematic process of operating, maintaining, upgrading, and disposing of assets cost-effectively



Picture 3. NFC Asset management system (NFC World, <http://www.nfc-world.com/en/cases/commodity.html>)

Tag management system, allows companies to use NFC tags to provide NFC-based asset management services to their existing customer base. NFC tags can be attached to physical objects and documents such as birth certificates, college diplomas, and certificates of authenticity for expensive or unique items, as well as high-value works of art, asset service history and warranty records. Authentication is often a problem in such

cases because of forgery and other frauds. Utilizing tag management system and NFC tags it's possible to avoid those risks.

Tag management system could be used alone or with existing asset management system. NFC tags could be used to label products in storages for tracking and inventory control. Another case is asset and user authentication. Benefits are following: user or asset identifies correctly excluding human error, information about the products retrieved in fast and easy manner.

4 Case study: TagMan - NFC Tag management system

This chapter is about TagMan NFC tag management system which was developed during 2012 - 2013 and implements features and ideas described in previous chapters.

4.1 TagMan overview

TagMan is NFC tag management system which includes database, mobile phone application, and web interface to create and modify data stored in database. TagMan NFC tags contain NDEF message with URI RTD NDEF record, different types of content such as web link, pictures, audio, video, text could be dynamically allocated to the tag. It is possible to define to show different content depending of day of week, part of the day e.g. morning or evening, certain period. Tags could be grouped into the tag groups and managed as one entity. Tags or tag groups could be linked to the location with GPS coordinates, which allows integrating location based services within tag management system. Tag tap history will show when and how many times tags was tapped by mobile phone or another NFC device.

Mobile phone application could be used to encode and insert NFC tags into TagMan system. It is developed for mobile phones with Android OS version higher than 2.2.3. It could also retrieve data about tag content from database through HTTP connection. TagMan system currently is in pilot phase, and was used in NFCP Global Summit which was held 20-22 March 2013 at London, UK.

4.2 TagMan Use cases

There are a lot of possible use case scenarios for TagMan system (Figure 8). NFC tag encoded and registered in TagMan system could be added to a poster or a magazine. By tapping the tag the user will be directed to the desired web page or discount coupon. Service provider can modify the content without a need of recoding or changing the tag.

By touching TagMan NFC tag on the timetable at bus stop or station it could be possible to get the local bus timetable into user's mobile phone. Instead of standing at the bus stop for the next 20 minutes, user could spent time at nearest cafe.

One area where TagMan NFC tags could be used is shopping. There are already some manufactures using UHF EPC tags in their products. EPC written to the tag could tell to the customer where the product was produced, other manufacturer information. For shop owners, it is also convenient to track products and check expiration date e.g. meat products. EPC code could be written to the NFC tag and attached to certain product, shelf or fridge. It also could provide additional services like audio information to visually impaired people, or information about product ingredients, which could be important to vegetarians, religious groups and persons with allergy.

TagMan NFC tag could be used at museums, to replace audio guides. Instead of pressing numbers to listen information about the piece of art or location, person just tap NFC tag and get video, text or audio. So there is not only one form of media could be used, but also other ones, increasing usability and customer satisfaction.

One of use case is home inventory. User could label boxes, shelves, containers with TagMan NFC tags. With the mobile phone he could easily locate things he searches at his home.

TagMan NFC tags could be used in asset management systems. It concerns not only tracking and inventory control in storages, but also at another premises like office, by labelling documents, certificates.

TagMan NFC tag embedded in business cards makes sharing of contact information really easy. No need to collect printed b-cards and type of contact information afterwards. If any information will be changed there is no need to throw away old business card with NFC tags, updated information will be available straight away after modifying data at TagMan system.

By tapping TagMan NFC tag on a Bluetooth device user's NFC smart phone will be automatically connected to the device. User can listen to his favourite music from BT speakers with only one touch. If other Bluetooth speaker or device will be used instead, user just needs to modify Bluetooth link at TagMan system. TagMan NFC tag provides an easy access to phone numbers. Local taxi number in hotel lobbies, restaurants or at office is handy. Tap the NFC tag and make a call.

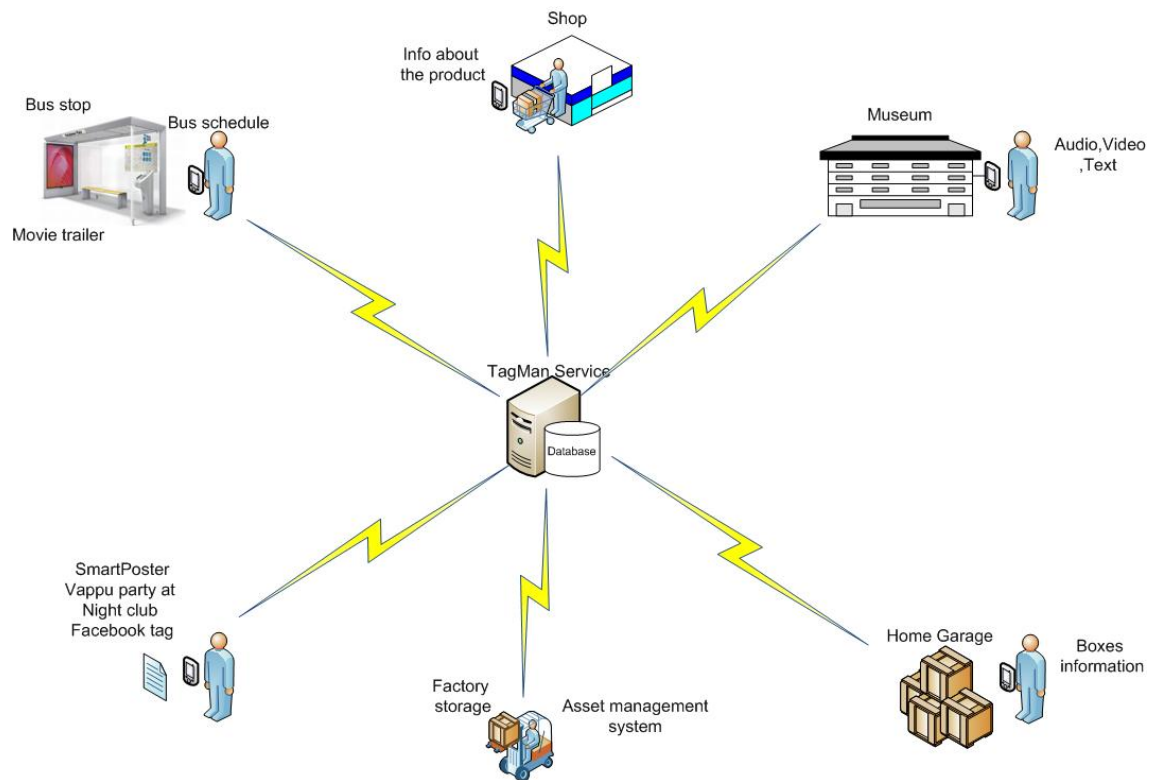


FIGURE 8. TagMan Use cases

TagMan NFC tag can be directed to activate a sending of a SMS message with a fixed content to a specified receiver number. User can buy e.g. a bus ticket and after sending the SMS, he'll get the ticket into his phone. In case of changes with ordering system like different number, where SMS should be sent or message structure, everything could be modified at TagMan system.

4.3 Development environment

Different tools and technologies were used during development of TagMan system. Web pages are located on Apache 2.2 web server. Apache is open-source HTTP server. It is recognized as the world's most popular web server. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Apache supports a variety of features, many implemented as compiled modules which extend the core functionality. These can range from server-side programming language support to authentication schemes. Some common language interfaces support Perl, Python, Tcl, and PHP.

PHP 5.2.1 was used to develop web pages. PHP is an open-source server-side scripting language designed for Web development to produce dynamic Web pages. It is one of the first developed server-side scripting languages to be embedded into an HTML source document rather than calling an external file to process data. The code is interpreted by a Web server with a PHP processor module which generates the resulting Web page. It has also evolved to include a command-line interface capability and can be used in standalone graphical applications.

As Database server MySQL 5.5 was chosen. MySQL is the world's most used open source relational database management system (RDBMS). MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack. LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

For mobile application development Android OS was chosen. Application was developed for Samsung Galaxy Mini 2 mobile phone, which has Android OS version 2.2.3. Android is a Linux-based operating system designed primarily for touchscreen mobile devices such as smartphones and tablet computers. Android is open source and Google releases the code under the Apache License. This open source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. Additionally, Android has a large community of developers writing applications that extend the functionality of devices, written primarily in a customized version of the Java programming language.

Server-side for communication with mobile phone was written on Java and located on Apache Tomcat 6 as Servlet. Apache Tomcat is an open source web server and servlet container developed by the Apache Software Foundation. Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run.

4.4 TagMan architecture

TagMan service consists of Trikker server and Android mobile phone application (Figure 9). Trikker server is name of ToP Tunniste application server where different services are located. Trikker server has Tomcat 5 servlet container, where servlet communicates with Android mobile phone application, it also retrieves and inserts data into MySQL database. Communication between Android application and servlet is done by HTTP POST request with JSON formatted strings.

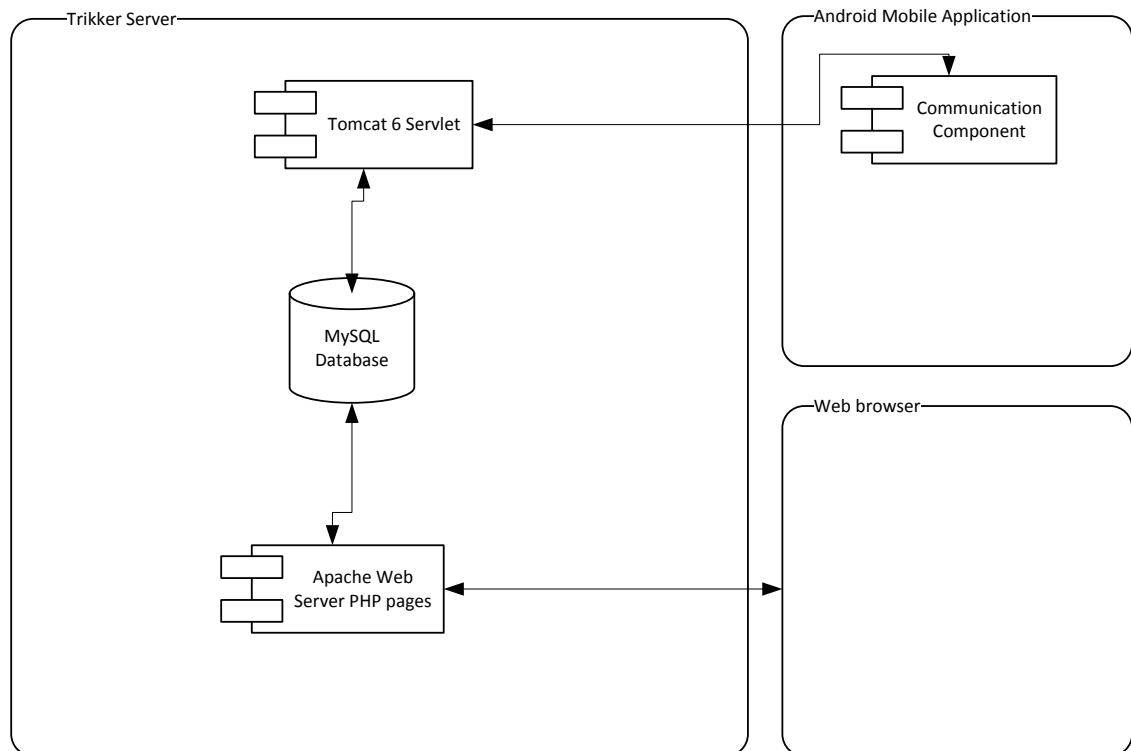


Figure 9. TagMan Service architecture

Android mobile phone application is implemented with Android version 2.2.3, it has capability to read and write NFC tags. NFC tags used in TagMan service are MIFARE Ultralight tags manufactured by UPM. There is NDEF message encoded on the tag, data itself is URL RTD.

There are PHP pages located at Apache 2.2 web server. User creates and modifies data located at MySQL database server by using web interface provided by PHP pages. PDO and Smarty web template system is used in PHP.

4.4.1 Database implementation

MySQL 5.5 is used as database server. MySQL Workbench tool was used to develop database and create SQL script. Database consists of twelve tables (Figure 10). User table represents user of the TagMan system. It has following fields: first name, last name, company, email, phone, password, those fields describe personal and contact information, email and password used to logging into the system. User level field shows access level of user. Everyone who uses the TagMan system should be registered and his or her profile is stored in this table.

Tag table describes NFC tag. It has following fields: tag UID, which is unique identifier of tag and created during manufacturing of the tag, tag created date (date then tag was inserted onto the system), tag type, tag group, location, default content, activity status and user to who tag belongs to.

Location table contains information about the physical location of tag (country, city, zip code, street address, province, location name, GPS coordinates, row creation date, and activity status). GPS coordinates such as longitude and latitude is FLOAT (10, 6) this was done to simplify integration with Google Maps service.

Content type table contains different types of data which tag could have. There are following content types: URL, text, picture, VCARD, SMS, phone call, email, sound, video, Facebook, Twitter, Foursquare, YouTube, Bluetooth, WIFI and coupon.

Content table contains data which tag could have. It has following fields: content type, content data (e.g. <http://www.toptunniste.fi>), content name, content description and user who have created this content.

Tag type contains different tag types. Currently there is NFC only, but in future also will be visual tags such as QR code. Barcode and UHF tags with EPC also could be added.

Tag group table describes group information. Tags could be united into big groups, this is done to simplify management of tags which have same data content and used in the same way.

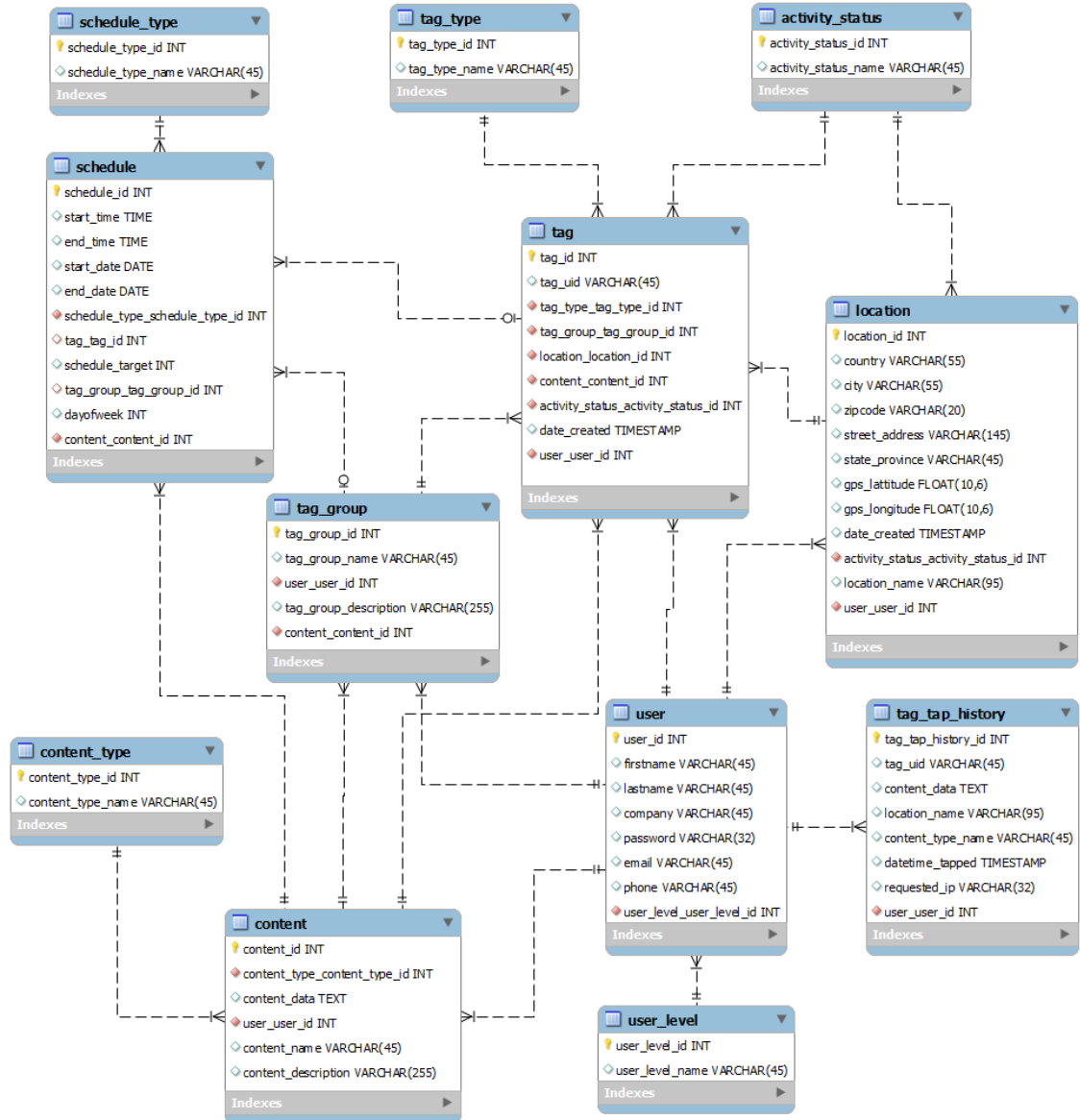


FIGURE 10. Database

User level type table describes different types of users. There are three levels of users which define access rights and other functionality. They are: user, administrator, and company.

Activity status table describes status of location and tag. There are two statuses active and inactive.

Schedule table is quite important entity, defining when certain tag content is shown to the user. It has following fields: start time, end time, start date, end date, tag, tag group, day of the week, content, schedule type and schedule target.

Schedule type table describes different schedule types. There are three schedule types: weekly, daily, period. Those three types define when tag content is shown.

Tag tap history table keeps data about the tag taps and content returned to the user of TagMan system. This data allows getting information about how often certain tag has been accessed by mobile device, at what time and what content has been shown to user. Table has following fields: tag, content, location, content type, date and time, IP address of mobile device.

4.4.2 Server-side architecture

Servlets are used to implement server side in order to communicate between mobile phone application and server and insert or retrieve data from database. Servlets are located at Tomcat 6 servlet container. There were different options to implement server side functionality, one of simplest way was to use PHP pages, second option was to make web service with Apache Axis 2 and third one is servlet. Due to usage of Tomcat in previous projects, servlets were chosen. A servlet is a Java programming language class used to extend the capabilities of a server. Although servlets can respond to any types of requests, they are commonly used to extend the applications hosted by web servers. In order to communicate with MySQL database JDBC driver for Tomcat called MySQL connector were used.

Communication between mobile phone and servlet was done using HTTP protocol, by POST requests. Data itself between mobile phone application and servlet is JSON formatted string. JSON or JavaScript Object Notation is a text-based open standard designed for human-readable data interchange. It is derived from the JavaScript scripting language for representing simple data structures and associative arrays, called objects. Despite its relationship to JavaScript, it is language-independent, with parsers available for many languages. In server side, Google Gson library was used to parse JSON strings. Google Gson is a Java library that can be used to convert Java Objects into their

JSON representation. It can also be used to convert a JSON string to an equivalent Java object.

To insert or retrieve data from MySQL, SQL queries were used. Another option which was considered is to use Hibernate. Hibernate is an object-relational mapping (ORM) library for the Java language, providing a framework for mapping an object-oriented domain model to a traditional relational database. But since database is not big enough and still a lot of changes will be in the future, it was decided to implement Hibernate mapping in the future releases.

4.4.3 Mobile application

Mobile phone application is important part of tag management system. During design of the system it was considered that special software is needed to encode tag and insert them into TagMan database. One of the options was to create desktop application. But then the process of encoding and inserting tags becomes too complicated. Mobile phone application was considered as the best option. More and more mobile phones with NFC capabilities appear at the market, most of smartphone users have internet subscription in the phone.

Next step was to select development platform to implement mobile application. In year 2012 the choice was between Android and Symbian, Windows Phone 8 platform with SDK was introduced at the end of the year 2012 and first Windows 8 phones came to market at the beginning of 2013. Symbian platform was the oldest which supported NFC, but since Nokia has decided to abandon it, Android was chosen as development platform. Android support of NFC functionality has started from version 2.2.3, that version was used to develop application so even the oldest Android phones which has NFC could install this software.

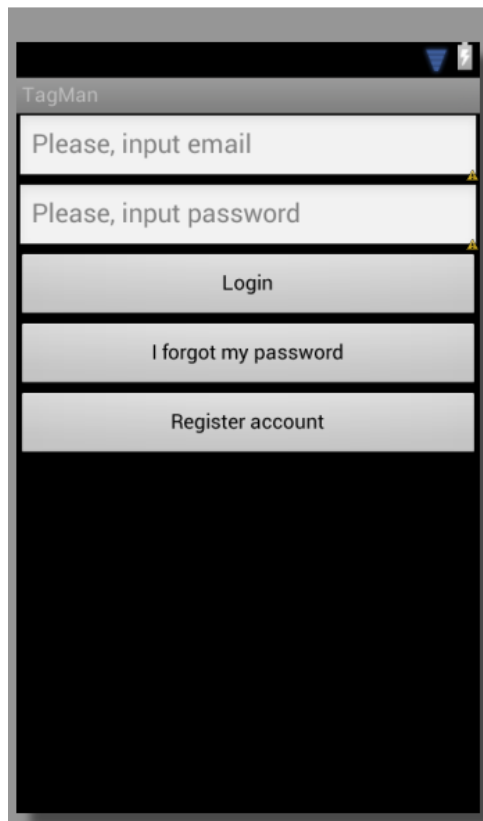


FIGURE 11. Login screen

First screen shown at Figure 11 is login screen, to use the system user should at first register at website, same email as login and password is used in phone application and in the web site. By pressing register account, user will be forwarded to registration web page. If user forgot password, he could press “I forgot my password” button and new password will be send to email. If login is successful user will be forwarded to next screen.

Three buttons for three options are shown at main screen (Figure 12). One of the options is to encode tag, by pressing it user will see screen shown at Figure 13. User should bring tag to the phone NFC antenna approximately from 3 to 5 centimetres distance; usually it located at back cover of the phone. It will take less than a second to encode data to the tag. After successful encoding, application will communicate to TagMan server, to insert encoded tag into database. If tag was encoded correctly and inserted the screen at Figure 14 will be shown.

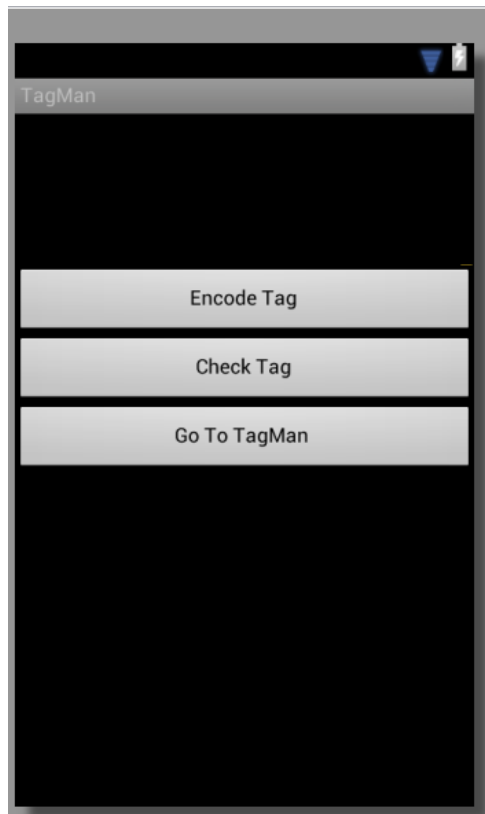


FIGURE 12. Main screen

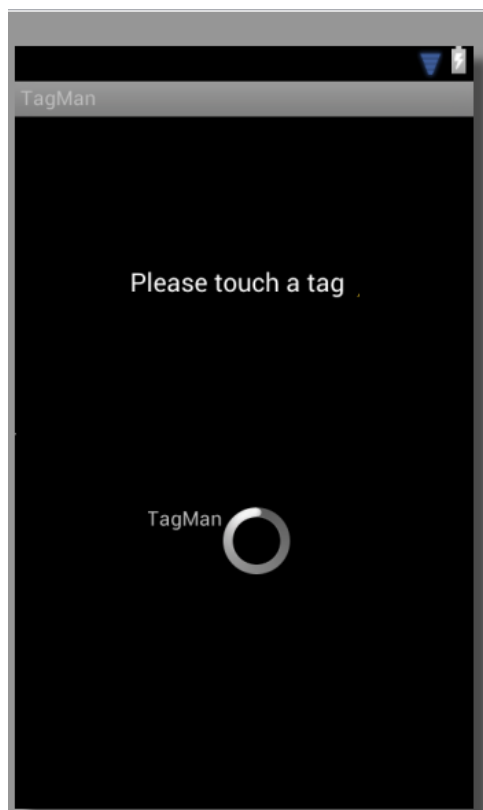


FIGURE 13. Touch tag screen

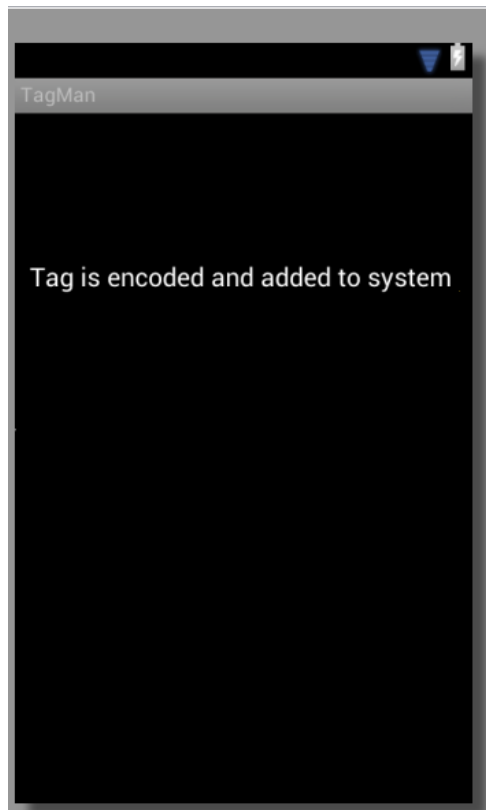


FIGURE 14. Tag inserted screen

Second option at main screen is to check NFC tag. Every user who use TagMan system has own tags, locations, tag groups and contents. To check whether tag is encoded or not and to get information about the tag user should press “Check tag” button. After pressing button, touch tag screen will appear, user should bring tag to close proximity to NFC antenna. Mobile application will connect to TagMan server and retrieve data about the tag. It’s important to say that only tags which belong to user could be checked, if tag belongs to another user, information about the tag won’t be provided. Information about the tag includes tag UID. Tag UID is unique identifier of NFC tag, for every tag type it could be different length, in documentation it is represented as hex number. Tag group describes tag group to which particular tag belongs. Content field shows name of content and content type which is currently linked to tag. Location point to certain place where tag is located or intend to locate (Figure 15).

Third option in main screen menu will open phone browser and forward user to TagMan web pages.



FIGURE 15. Tag information screen

4.4.4 Web interface

Web pages are the most important part of TagMan system. Most of the time owner of tags will spend creating, modifying data at web pages. The core functionality is implemented there. ToP Tunniste has registered tagm.eu domain specifically for TagMan system. First page user will see after typing tag.eu URL in web browser will be login page (Figure 16). If login is successful user will be forwarded to the next page, if login failed user will be asked again to login with his credentials. Before starting using the system user should be registered. If user is not registered he could do it by pressing “Register” link at login web page. Then register web page will be shown to the user (Figure 17).

Register page contains fields which must be filled in order to register in TagMan system. There are following fields: first name, last name, company (this field is not com-

pulsory), email, which will be user as login name, password, and second password, which must be identical to first one, phone number.

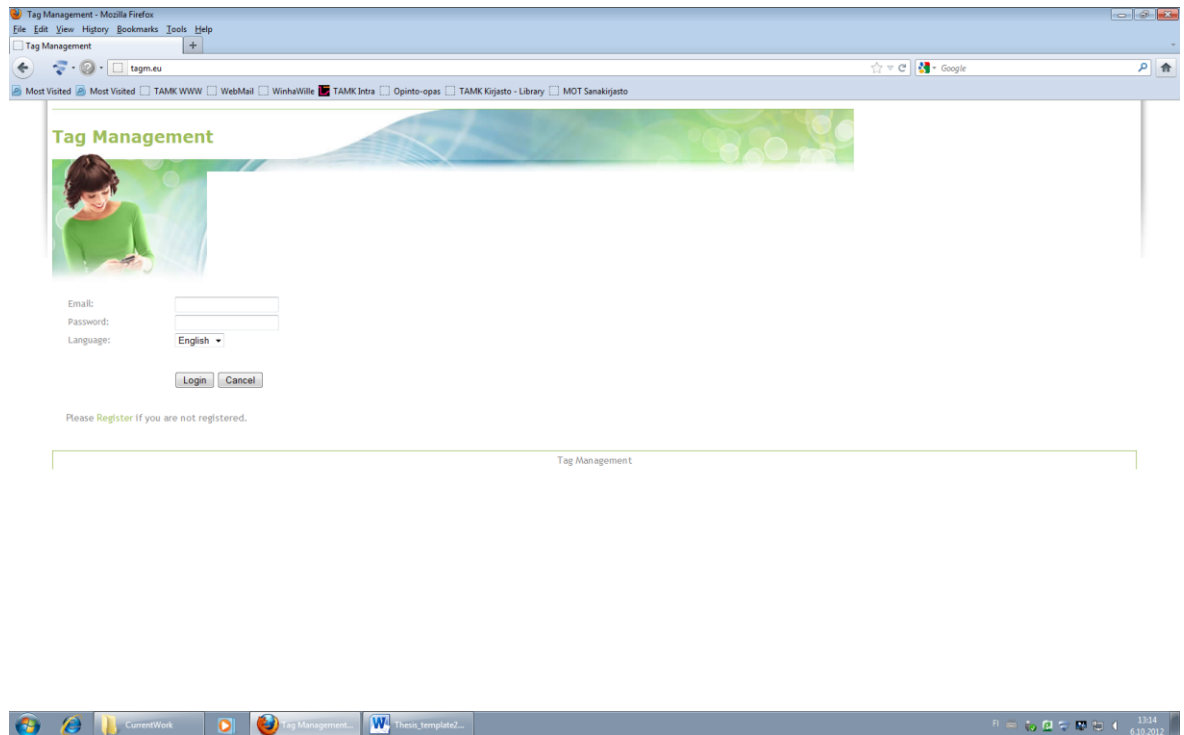


FIGURE 16. Login web page

After filling registration page, user must press “Save” button. If information in the fields is correct and user has filled all necessary fields, user will be relocated to login page. If information is wrong or incomplete, user will be asked to do it again.

After successful login user will go to tag page. At this page all available user’s NFC tags are listed (Figure 18). In the table there are information about tag group, tag location, tag UID, tag content and button to edit tag.

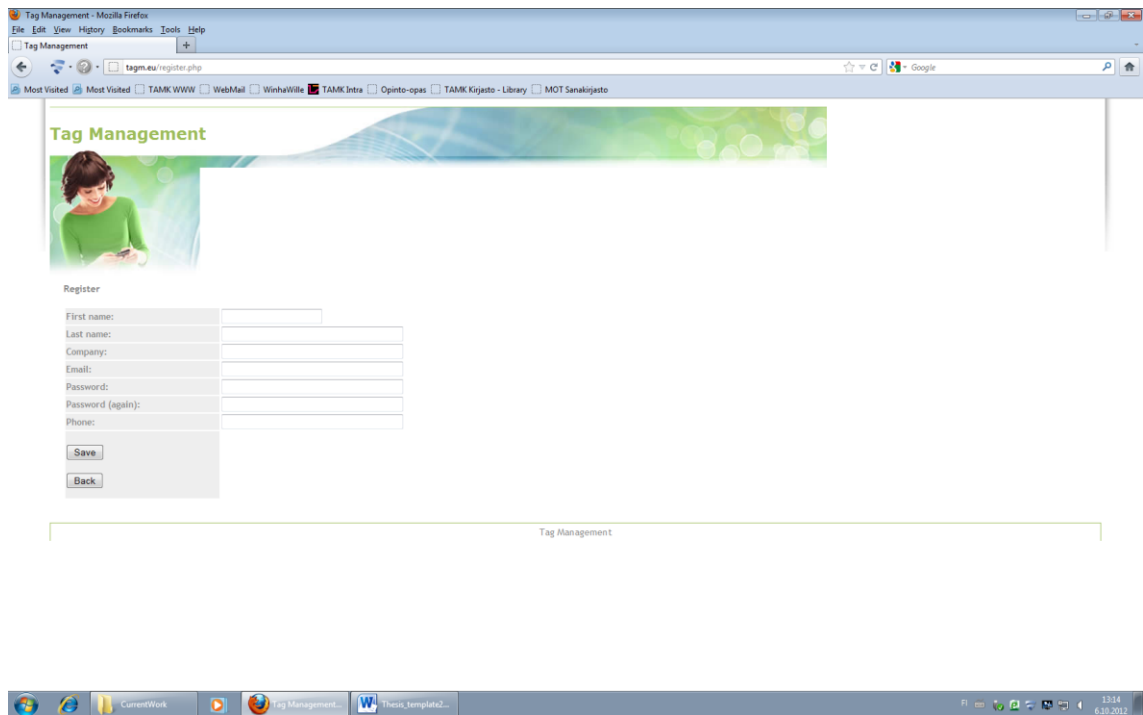


FIGURE 17. Registration web page

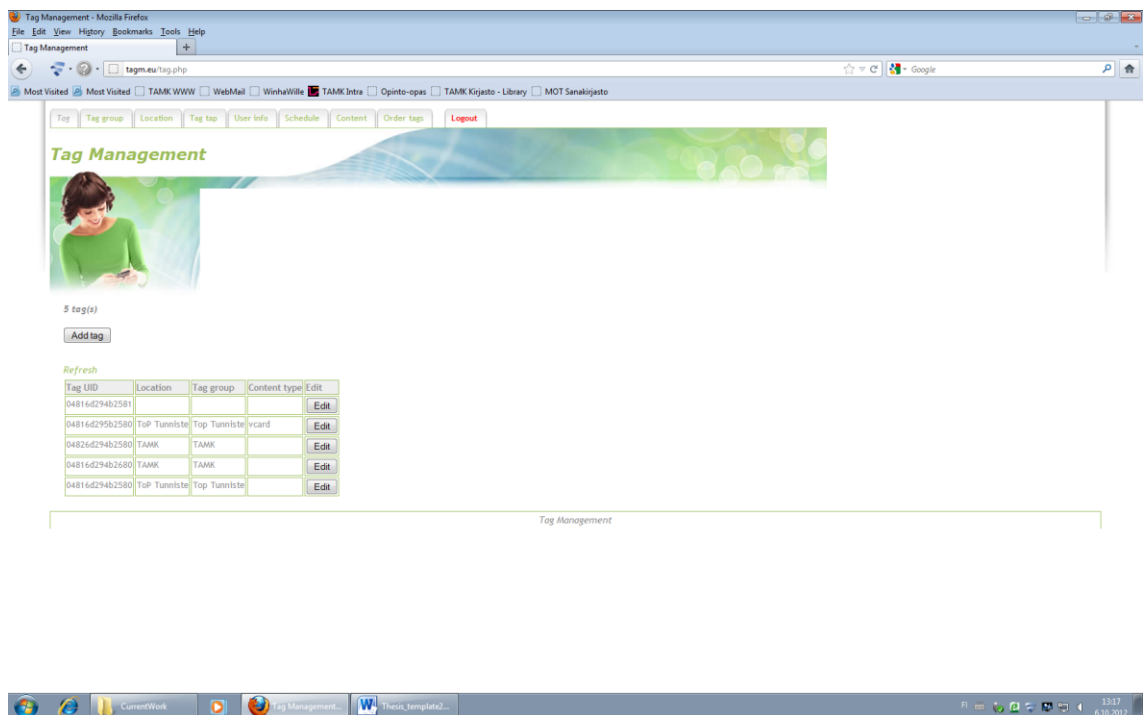


FIGURE 18. Tags web page

Tag edit or add page shown at Figure 19 allows modify tag data or create brand new tag. User could not set tag UID to the tag which belongs to another user; otherwise there is a security breach, which could be used to acquire control on another user's tags. It is pos-

sible to set tag group to certain one or user could left tag group not selected then tag won't be in any group. User could set tag content to the tag; it will be default content which will be shown to person who is tapping tag at location. User also can set tag location or left it not selected. If user tag has location, map from Google will be visible at web page. This functionality is implemented with Gmap3. Gmap3 is a jQuery plug-in to create Google maps with advanced features. jQuery is a multi-browser JavaScript library designed to simplify the client-side scripting of HTML. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library and Gmap3 one of these plug-ins.

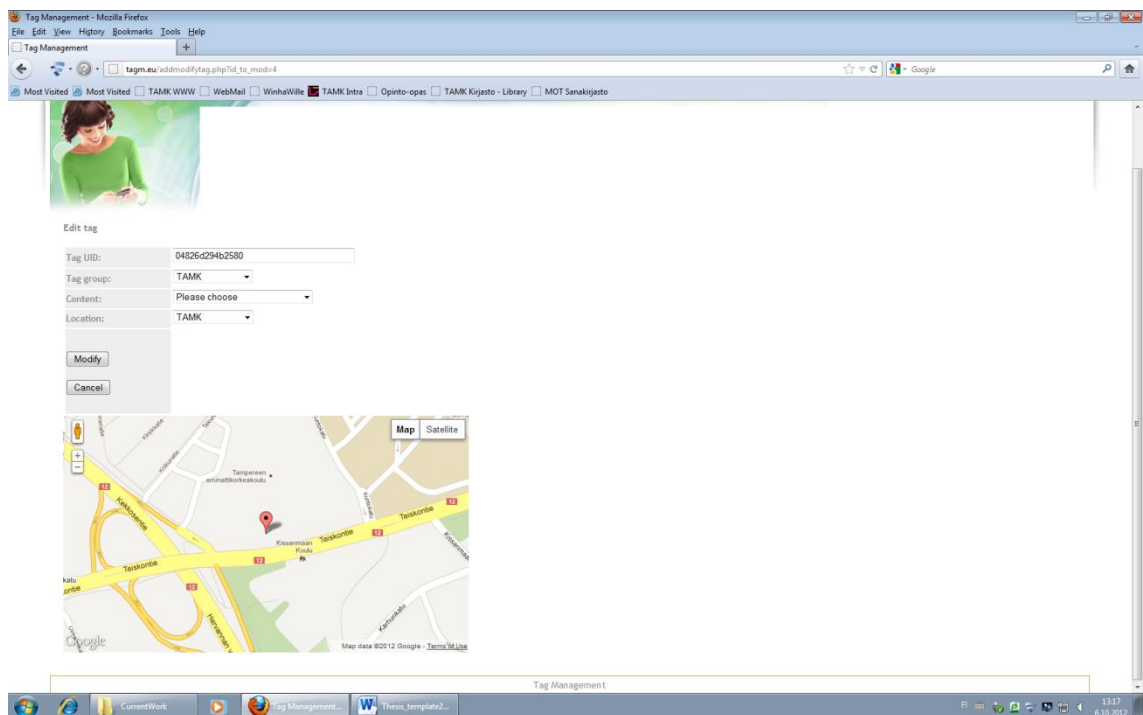


FIGURE 19. Tag edit or add web page

Second tab at the top menu is tag group. Tag group web pages lists tag groups information (Figure 20). There following columns in the table: tag group name, tag group description, amount of tags and edit button to modify tag group information.

Tag group edit web page allows add or edit tag group (Figure 21). Tag group name field is used to give a certain name to the tag group. Tag group description text field contains description of tag group, it should be short, and maximum allowed number of characters is 255.



FIGURE 20. Tag group web page

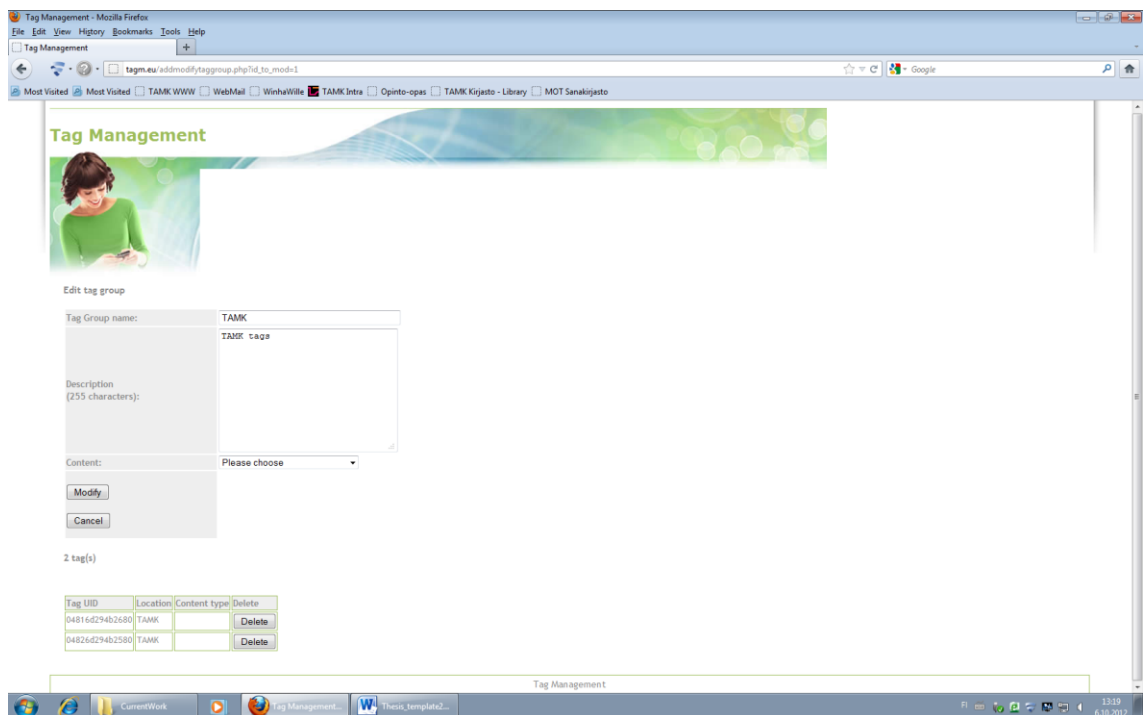


FIGURE 21. Tag group edit or add web page

User also could choose content which will set as default content to the tag group. If there is no schedule linked to the tag group, this content will be shown to the user who

taps the tags which are in this tag group. Content could be left empty then TagMan system will show information about the TagMan system or advertisement.

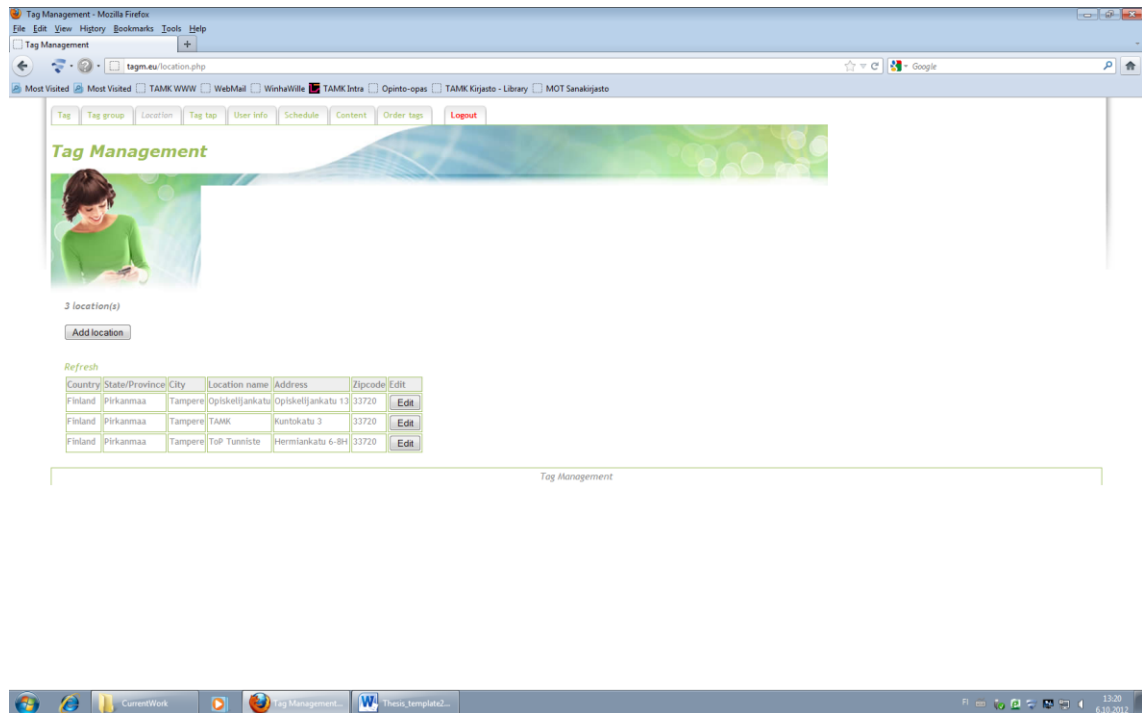


FIGURE 22. Location web page

Location web page shows information about users' locations (Figure 22). There are following fields in the table: country, state or province, city, location name, address and zip code. Those locations are used to link them to certain tag. To create new location or edit existing one, user could press 'Edit' button or 'Add location'.

Location edit page contains fields to edit or create new location (Figure 23). There are same fields which are shown in location list at location web page. Also there are two additional fields which are latitude and longitude. Those fields are used to show map at the bottom of edit fields. To create Google map, Gmap3 jQuery plug-in is used.

Tag taps web page shows tag tapping details (Figure 24). Table has following columns: tag UID, location, content type, content, date and time when tag was tapped. This information allows to see how many times and exact date and time, when some person at certain location has tapped tag to get information or request service from the tag.

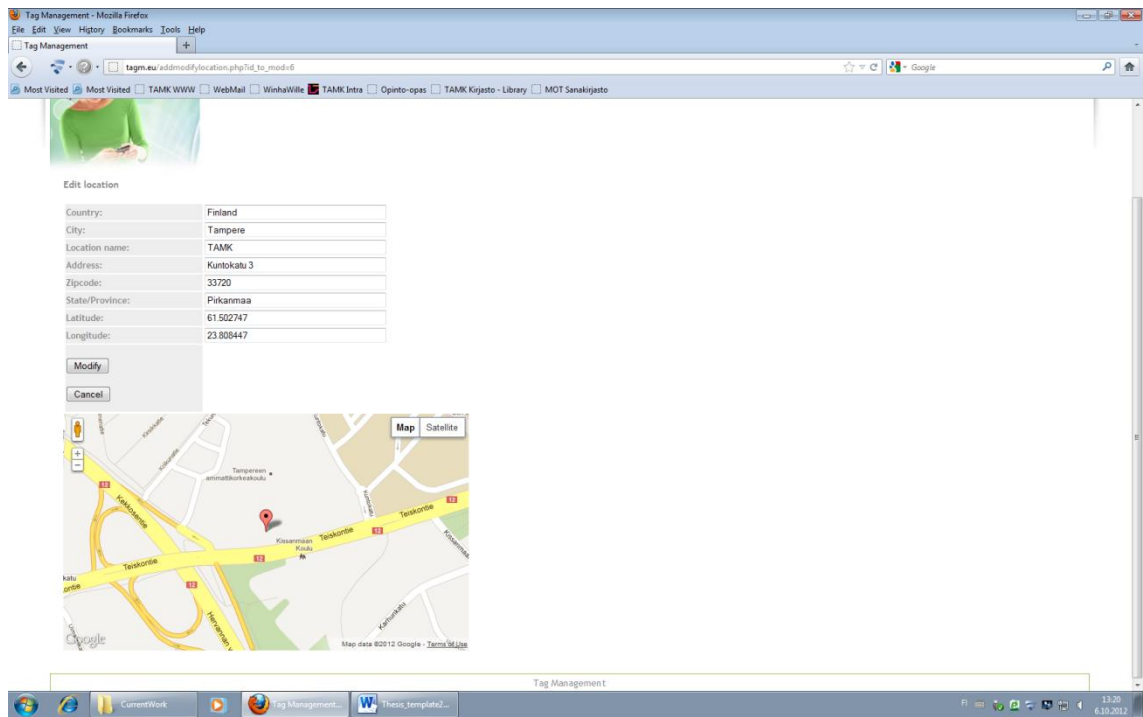


FIGURE 23. Edit or add location web page

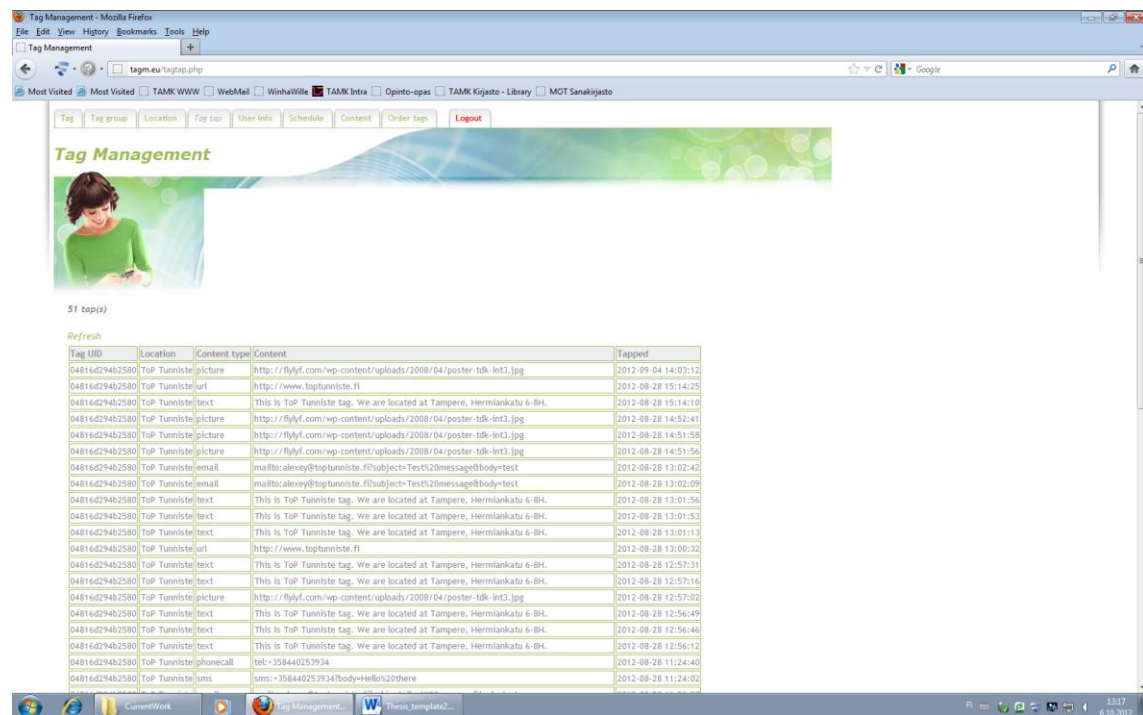


FIGURE 24. Tag taps web page

Tag schedule web page shows schedule list for certain tag or tag group (Figure 25). Schedule could be done for certain tag or for tag group, if tag UID field has value it means that schedule is done for this tag, and if tag group field has name then it is for tag

group. Schedule type field defines when certain tag content will be shown to user who taps tag.

Currently there are three schedule types; tag content could be show daily from certain start time till certain end time, weekly, every Monday or another day of the week, or third option by certain period. Content field contain content name which will be shown during this schedule is active. Start time and end time fields are used if daily or period schedule type is selected. Day of the week filed is used if weekly schedule type is used. Start date and date fields used only if period schedule type is selected.

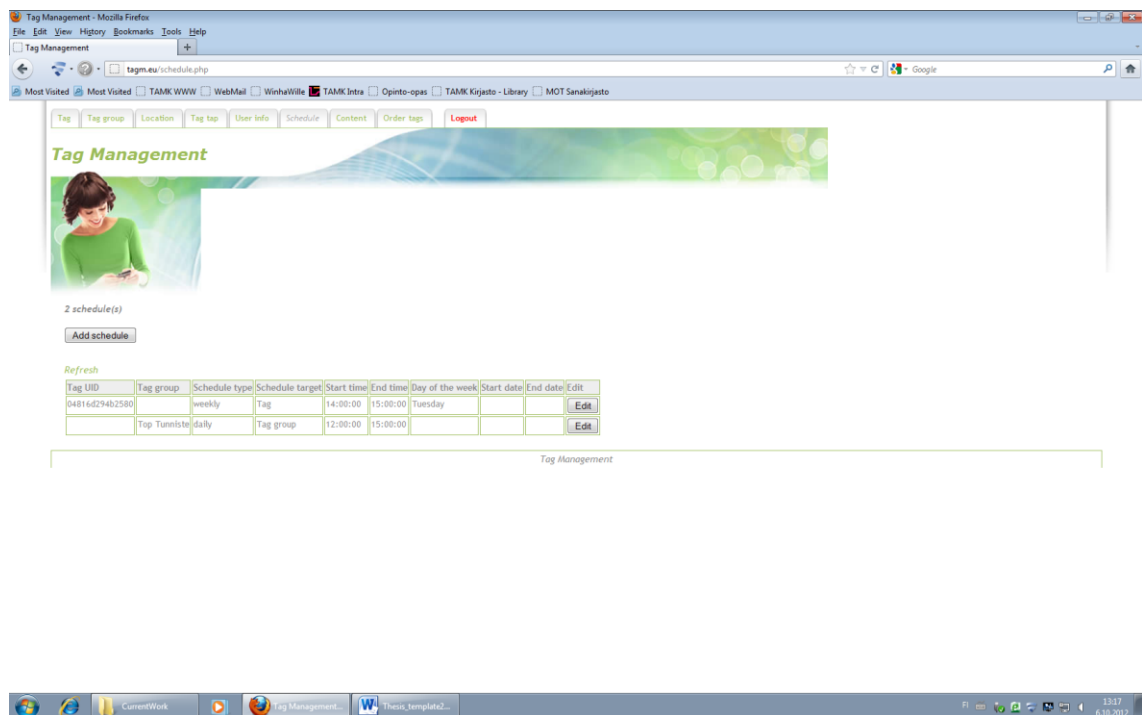


FIGURE 25. Schedules web page

There is schedule add, modify web page where user could change these values (Figure 26). For one tag or tag group it could be done many schedules. If these schedules are conflicts or overlaps the last inserted schedule is used.

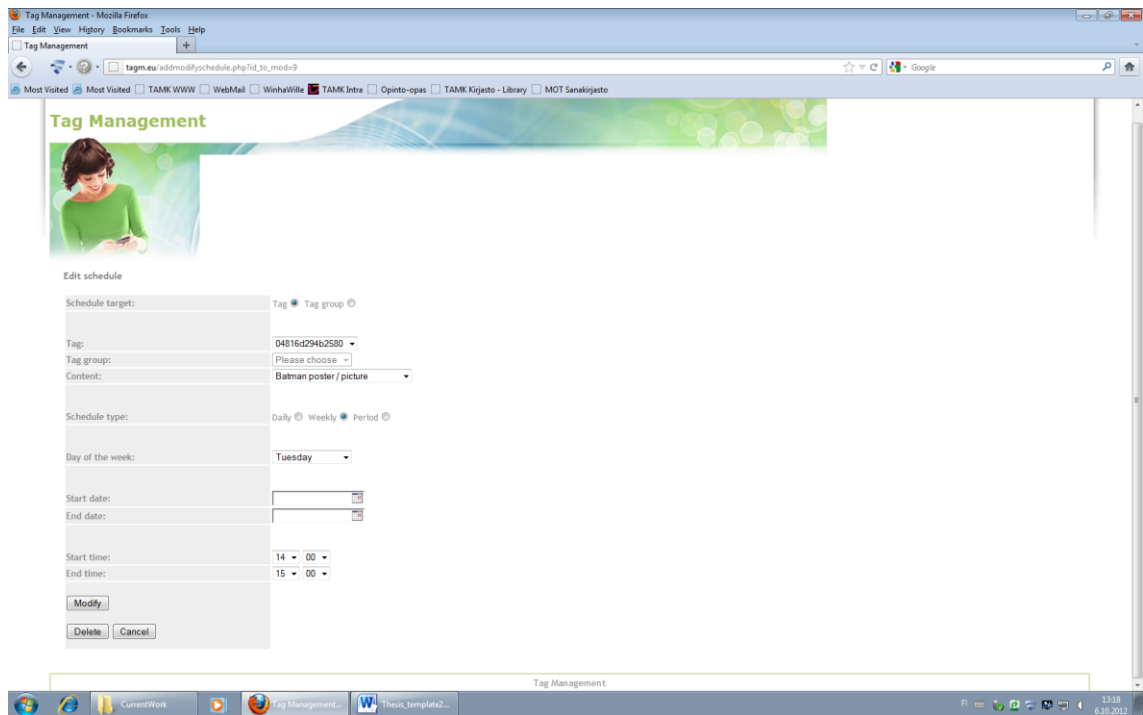


FIGURE 26. Edit or add schedules for tag web page

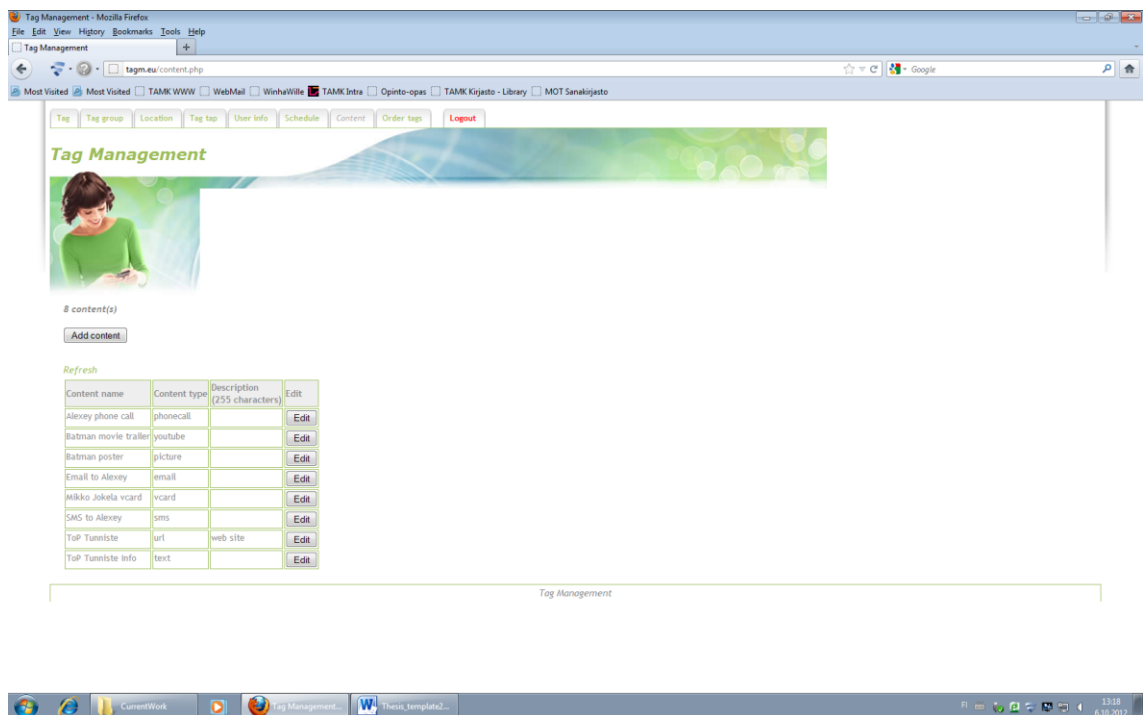


FIGURE 27. Edit or add schedules for tag groups web page

Tag content web page shows different contents created by user (Figure 27). Content table contains content name field, content type, description which could be up to 255

characters long. User could press ‘Edit’ button to modify content or ‘Add content button’ to create new one.

Tag content add or modify web page is shown at Figure 28. Content name field contains name of the content, description text area describe content, and is not compulsory, content type select has different types.

Now there are following content types: URL, text, picture, VCARD, SMS, phone call, email, sound, video, Facebook, twitter, foursquare, YouTube, Bluetooth, WIFI and coupon. For example user could create URL to the TAMK canteen, in the content field name will be ‘TAMK ruokalista’, content type is URL, and content field ‘<http://www.campusravita.fi/index.php?id=2&week=true>’. So we created a URL content which shows canteen menu for a week. Same way we could create phone call to support service or SMS to pay for a parking.

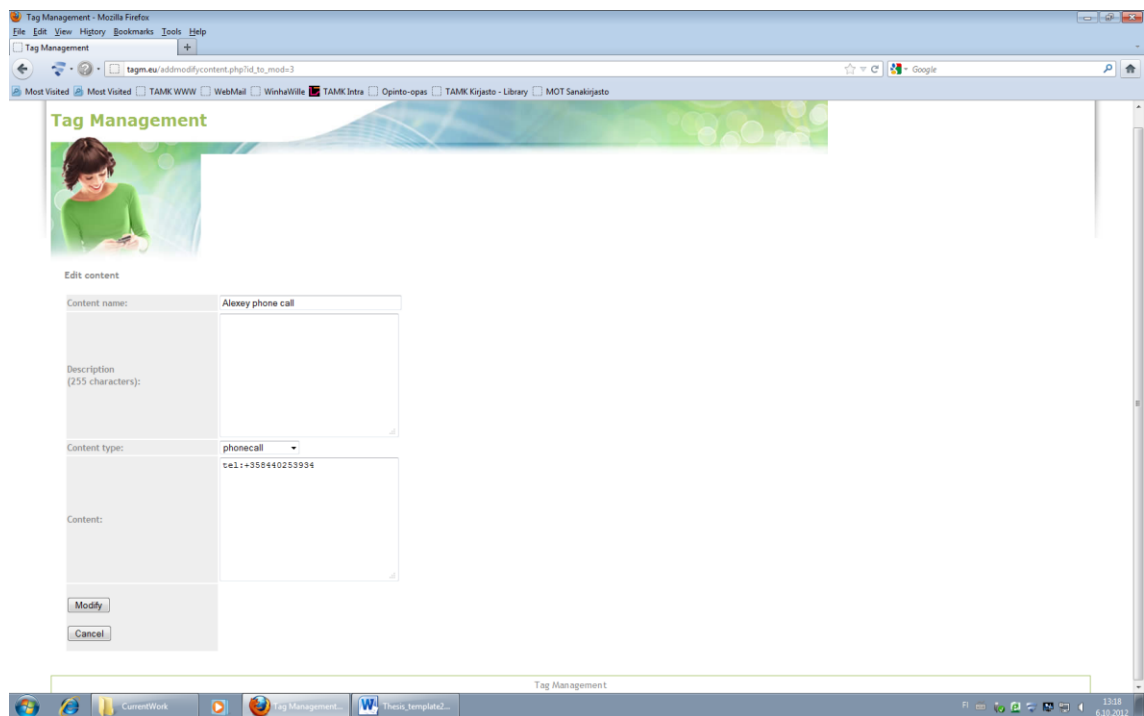


FIGURE 28. Edit or add schedules for tag groups

4.4.5 TagMan NFC Tag

Last but not the least part of TagMan system is NFC tag (Figure 29). There are many types of NFC tags with different features and capacity. It was necessary to select certain type of tags which is widely used and not expensive. Nowadays the MIFARE Classic 4k (4 kilobytes) and MIFARE DESFIRE 8K (8 kilobytes) are tags which have the biggest capacity. But since MIFARE Classic is not NFC type tag, though it's supported by most of NFC readers and phones, those type of tag was not selected, due to possible compatibility problems. MIFARE DESFIRE tags was also rejected because they are not widely used and expensive. So finally choice was between Innovision TOPAZ tags and MIFARE Ultralight tags. Innovision TOPAZ tags has one advantage over MIFARE Ultralight tags, they offer bigger memory capacity. On other hand we don't need a lot of memory, so MIFARE Ultralight tags was selected due to cheaper price and wide support of NFC manufacturers.

Second step was to select RTD type for NDEF record which will be used in NDEF message encoded to the tag. It is possible to include many NDEF records in NDEF message, which have different RTD types, e.g. URI and external RTD or MIME type. It's done to make possible to auto launch, not only web browser but certain application to process NDEF message. There is only one drawback of having two NDEF record in NDEF message, it is interoperability problem. For example during testing it was revealed that Android could auto launch application by MIME type, but not External RTD. Windows 8 phones has their own proprietary NDEF Record to auto launch application and it not supports neither MIME RTD nor External RTD. It was decided that URI RTD will be used in NDEF message. The content of URI is "http://tagm.eu/?uid", where UID is unique NFC tag UID.



FIGURE 29. TagMan NFC tag

4.5 Future Development

One of the things which should be developed further is web pages. For end user especially for companies it's important to know statistics, how many times their tags were tapped. Detailed statistics with graphic charts should be added to tag taps web pages. Now only NFC tags are supported in TagMan, adding support for visual tags, barcodes and UHF tags with EPC code will attract more users and companies which are using these technologies. Support for different content types is also one of goals for future development. For example Facebook 'Like' tags definitely will be hardly demanded by common users and companies. Another feature which could be added is sharing tags for different users, so one tag at certain location could be used not only by one user. This will add flexibility and will allow using same tag in different time periods by different users. Additional features like advertisement loop is planned to be implemented. Advertisement loop consists of sequence of different web links, images, videos which are constantly changing every time then user taps the tag.

Currently mobile application is developed only for Android, support of other platform is another goal for future development. Since Windows Phone 8 platform gaining popularity, application for this platform is top priority. Also functionality of mobile application should be extended. It would be perfectly if user could do all the same actions he could do with web pages in mobile application.

There are some changes are planned to do at server side. One of changes will be to use Hibernate for mapping an object-oriented domain model to a traditional relational database which is MySQL in our case. It helps to get rid of writing complex SQL statement, to reduce the lines of code and emphasizes more on business logic, to abstracts application away from the underlying SQL database and SQL dialect.

5 Results and conclusions

The results of this Master's thesis could be divided in two parts: NFC tag management system and tag management methodology. The TagMan tag management system proof of concept that was implemented as a part of this Master's thesis can already be used for handling of NFC tags. TagMan database, web pages, mobile application, server side and tag content was designed and implemented and was taken into use. It was tested at NFCP Global Summit at London and currently it is still in pilot phase.

Tag management methodology including models and concepts is another result. Different ways of identifying objects was researched from old barcode technology to modern NFC. Important conclusion was suggesting solution of combining QR code, NFC tag and EPC code in one object, which could be used for many services during whole life cycle of object from creation to utilisation and destruction. Statically and dynamically changed content was compared and admitted advantages of dynamically content over static. Hence it was proved necessity of tag management system.

Another tag management concept that was presented is one tag many services. One tag could belong not to only one owner but shared among many ones. It will allow to use same tag for many purposes from simple data retrieval to field force reporting. This will allow to use same NFC tags by many companies and reduce entropy in tag ecosystem.

Another result is researching business aspects of NFC tag management system and describing scenarios of integrating it with asset management system, field force solutions and location based services. Hence TagMan has big commercial potentials. It is perfectly fit for advertising campaigns and other commercial infotainment services.

Security aspects and ways of preventing misuse and frauds is another result in this Master's thesis. It was described typical security issues for IT and some unique for NFC tag management system. Protection solution against vandalism was presented developed by ToP Tunniste for RMV at Frankfurt and which also could be used for TagMan.

6 Summary

Near Field Communication is a new technology and ecosystem that has emerged in the last decade. Potential NFC applications and services making use of NFC technology include e-payment, e-ticketing, loyalty services, identification, access control, content distribution, smart advertising, data/money transfer and social services. Usability issues and technology adoption are being explored by many academicians and industrial organizations. As NFC enabled mobile phones spread and commercial services are launched, people will be able to pay for goods and services, access hotel rooms or apartments, update their information in social networks, upload their health data to hospital monitoring systems from their homes, and benefit from many more services by using their NFC enabled phones. NFC tags are one important part of NFC ecosystem and there is a need for tag management system which will handle those tags.

In this Master's thesis we described RFID technology and history of NFC which is based on it. It was defined reasons why NFC technology was demanded. Internet of objects and physical browsing as a new intuitive human computer interfacing paradigm for mobile users and its relation with NFC was shown. Different use case for NFC such as payment, ticketing and infotainment services was revealed. NFC architecture and its essential elements like three modes of NFC, NFC Data exchange format, Record type definition and NFC forum tag types was described.

We overviewed different NFC devices, NFC development tools and environments. Security and other issues related to NFC were discussed. Smart posters and tag emulation as ways to share data were compared in this work. Models and concepts of tag management system, use case scenarios and business aspects were researched.

Finally TagMan system as case study of tag management system was introduced and described. Architecture and implementation of database, web pages, mobile application and tag content was reviewed.

REFERENCES

1. RFID Sourcebook, Sandip Lahiri, Prentice Hall PTR, 2005, ISBN: 0-13-185137-3
2. RFID Security, Frank Thornton, Brad Haines, Syngress Publishing, Inc., 2006, ISBN: 1-59749-047-4
3. Near field communication : from theory to practice / Vedat Coskun, Kerem Ok, and Busra Ozdenizci, John Wiley & Sons Ltd, 2012, ISBN: 9781119971092
4. NFC Data Exchange Format (NDEF) Technical Specification NFC Forum NDEF 1.0 NFCForum-TS-NDEF_1.0
5. NFC Record Type Definition (RTD) Technical Specification NFC Forum RTD 1.0 NFCForum-TS-RTD_1.0 2006-07-24
6. URI Record Type Definition Technical Specification NFC Forum RTD-URI 1.0 NFCForum-TS-RTD_URI_1.0 2006-07-24
7. Text Record Type Definition Technical Specification NFC Forum RTD-Text 1.0 NFCForum-TS-RTD_Text_1.0 2006-07-24
8. Smart Poster Record Type Definition Technical Specification NFC Forum SPR 1.1 NFCForum-SmartPoster_RTD_1.0 2006-07-24
9. Signature Record Type Definition Technical Specification NFC Forum SIGNATURE 1.0 NFCForum-TS-Signature_RTD-1.0 2010-11-18
10. Type 1 Tag Operation Specification Technical Specification NFC Forum T1TOP 1.1 NFCForum-TS-Type-1-Tag_1.1 2011-04-13
11. Type 2 Tag Operation Specification Technical Specification T2TOP 1.1 NFC Forum NFCForum-TS-Type-2-Tag_1.1 2011-05-31

12. Type 3 Tag Operation Specification Technical Specification NFC Forum
T3TOP 1.1 NFCForum-TS-Type-3-Tag_1.1 2011-06-28
13. Type 4 Tag Operation Specification Technical Specification NFC Forum
T4TOP 2.0 NFCForum-TS-Type-4-Tag_2.0 2011-06-28
14. Touch the Future with a Smart Touch. Eds. Tuomo Tuikka & Minna Isomursu.
Espoo 2009. VTT Tiedotteita – Research Notes 2492.
15. Mobile NFC Technical Guidelines – vs2 GSMA
16. RFID For Dummies, 2005, Published by Wiley Publishing, Inc., Indianapolis,
Indiana
17. GS1 EPC Tag Data Standard 1.6, EPCglobal
18. Ernst Haselsteiner, Klemens Breitfuß: Security in near field communication
(NFC) PDF (158 kB), Philips Semiconductors, Printed handout of Workshop on
RFID Security RFIDSec 06, July 2006
19. Smart Posters. How to use NFC tags and readers to create interactive experi-
ences that benefit both consumers and businesses. April 2011. NFC Forum.
20. Kevin Ashton: That 'Internet of Things' Thing. In: RFID Journal, 22 July 2009.
21. <http://www.nfc-forum.org>
22. [http://msdn.microsoft.com/en-
us/library/windowsphone/develop/jj207060%28v=vs.105%29.aspx](http://msdn.microsoft.com/en-us/library/windowsphone/develop/jj207060%28v=vs.105%29.aspx)
23. http://en.wikipedia.org/wiki/Near_field_communication
24. http://en.wikipedia.org/wiki/Apache_HTTP_Server

25. <http://en.wikipedia.org/wiki/PHP>
26. <http://en.wikipedia.org/wiki/MySQL>
27. http://en.wikipedia.org/wiki/Android_%28operating_system%29
28. http://en.wikipedia.org/wiki/Apache_Tomcat
29. <http://gmap3.net/en/>
30. <http://en.wikipedia.org/wiki/JQuery>
31. http://en.wikipedia.org/wiki/Java_Servlet
32. <http://en.wikipedia.org/wiki/JSON>
33. http://en.wikipedia.org/wiki/Hibernate_%28Java%29
34. <http://en.wikipedia.org/wiki/Smarty>
35. <http://www.nfc-world.com/en/cases/commodity.html>
36. http://en.wikipedia.org/wiki/Asset_management
37. http://en.wikipedia.org/wiki/Denial-of-service_attack
38. <http://en.wikipedia.org/wiki/Barcode>
39. http://en.wikipedia.org/wiki/QR_code
40. http://en.wikipedia.org/wiki/Electronic_Product_Code
41. <http://ttuki.vtt.fi/smarttouch/www/?info=case-smartparking>

APPENDICES

Appendix 1. Add modify location PHP (addmodifylocation.php)

```
<?php
session_start();
$active_page = "location";
require("validation.php");
require("database_settings.php"); // database, tables, login, password etc.

// open database
$link = mysql_connect($host, $login, $password) or die('Could not connect: '
. mysql_error());
mysql_select_db($database) or die('Could not select database' .
mysql_error());

// Main page content begin

$country=$_POST['country'];
$city=$_POST['city'];
$locationName=$_POST['locationName'];
$address=$_POST['address'];
$zipcode=$_POST['zipcode'];
$stateProvince=$_POST['stateProvince'];
$gpsLongitude=$_POST['gpsLongitude'];
$gpsLatitude=$_POST['gpsLatitude'];
$id_to_mod=$_POST['id_to_mod'];

switch ($_POST['action']) {

    // add location
    case "._SAVE." :
        if ( $city != "" AND !is_null($city)
            AND $locationName != "" AND !is_null($locationName)
            AND $address != "" AND !is_null($address)

        ) {

            addData($country,$city,$locationName,$address,$zipcode,
                $stateProvince,$gpsLongitude,$gpsLatitude);

            // if fields not defined
        } else {

            generatePage();
            validate_data( $city,$locationName,$address);

        }
        break;

    // modify location
    case "._MODIFY." :

        if ($city != "" AND !is_null($city)
```

```

        AND $locationName != "" AND !is_null($locationName)
        AND $address != "" AND !is_null($address)

    ) {

        modifyData($country,$city,$locationName,$address,$zipcode,
        $stateProvince,$gpsLongitude,$gpsLatitude,$id_to_mod);

        // if fields not defined
    } else {

        generatePage();
        validate_data( $city,$locationName,$address);

    }
    break;

    case ""._CANCEL."":

        header("Location: location.php");
        break;

    case ""._DELETE."":

        deleteData($id_to_mod);
        break;

    default :

        generatePage();

}

// Main page content end

echo "\n";
echo "</div>\n"; // content end
echo "</div>\n"; // wrapper-content end

require("footer.php");

/**
 * Generate fields to insert the data
 * @access <public>
 *
 *
 * @return <none>
 */

function generatePage() {

    require("header.php");
    echo "<div id=\"wrapper-content\">\n";
    echo "<div id=\"content\">\n";

```

```

        echo ("<form method=\"post\" action=\""$PHP_SELF\"
name=\"infoForm\" >");
        echo ("<p><a href=\""$PHP_SELF"."\">". "</a>");

        // if modify
        if (isset ($_REQUEST['id_to_mod'])) {

            $id_to_mod = $_REQUEST['id_to_mod'];

            $sql = "SELECT country,city,location_name,street_address,
zipcode,state_province,gps_longitude,gps_latitude
FROM location WHERE location_id = $id_to_mod AND user_user_id = ".
$_SESSION['verified_user_id'] ."";

$result = mysql_query($sql);
$rows = mysql_num_rows($result);

if($rows != 0){

    echo "<h3>".$_EDITLOCATION."</h3>\n";

    $country = mysql_result($result, 0, 0);
    $city = mysql_result($result, 0, 1);
    $locationName = mysql_result($result, 0, 2);
    $address = mysql_result($result, 0, 3);
    $zipcode = mysql_result($result, 0, 4);
    $stateProvince = mysql_result($result, 0, 5);
    $gpsLongitude = mysql_result($result, 0, 6);
    $gpsLatitude = mysql_result($result, 0, 7);

}else{

    echo "<div class='error'>".$_NOSUCHLOCATION."</div>";

}

}else {

    echo "<h3>".$_ADDLOCATION."</h3>\n";

    // get previous values of variables if some error was done
    if (!isset ($_REQUEST['country'])) {
        $country = "";
    } else {
        $country = $_REQUEST['country'];
    }
}

```

```

        if (!isset ($_REQUEST['city'])) {
            $city = "";
        } else {
            $city = $_REQUEST['city'];
        }
        if (!isset ($_REQUEST['locationName'])) {
            $locationName = "";
        } else {
            $locationName = $_REQUEST['locationName'];
        }
        if (!isset ($_REQUEST['address'])) {
            $address = "";
        } else {
            $address = $_REQUEST['address'];
        }

        if (!isset ($_REQUEST['zipcode'])) {
            $zipcode = "";
        } else {
            $zipcode = $_REQUEST['zipcode'];
        }
        if (!isset ($_REQUEST['stateProvince'])) {
            $stateProvince = "";
        } else {
            $stateProvince = $_REQUEST['stateProvince'];
        }
        if (!isset ($_REQUEST['gpsLongitude'])) {
            $gpsLongitude = "";
        } else {
            $gpsLongitude = $_REQUEST['gpsLongitude'];
        }
        if (!isset ($_REQUEST['gpsLatitude'])) {
            $gpsLatitude = "";
        } else {
            $gpsLatitude = $_REQUEST['gpsLatitude'];
        }
    }

    echo("<td>");
    echo("<table border='0' width='60%' class='admin_form'>" . "\n";
    echo("<tr><th class='admin_form' nowrap>._COUNTRY.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='country' value='$country'></td></tr>");
    echo("<tr><th class='admin_form' nowrap>._CITY.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='city' value='$city'></td></tr>");
    echo("<tr><th class='admin_form' nowrap>._LOCATIONNAME.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='locationName' value='$locationName'></td></tr>");
    echo("<tr><th class='admin_form' nowrap>._ADDRESS.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='address' value='$address'></td></tr>");
    echo("<tr><th class='admin_form' nowrap>._ZIPCODE.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='zipcode' value='$zipcode'></td></tr>");
    echo("<tr><th class='admin_form' nowrap>._PROVINCE.':</th><td colspan='3' class='admin_form'><input type='text' size=40 maxlength=45 name='stateProvince' value='$stateProvince'></td></tr>");

```



```

        echo ("<tr><th class='admin_form' nowrap>"._LATITUDE."</th><td
colspan='\"3\"' class='admin_form'><input type='text' size=40 maxlength=45
name='gpsLatitude' value='$gpsLatitude'></td></tr>");
        echo ("<tr><th class='admin_form' nowrap>"._LONGITUDE."</th><td
colspan='\"3\"' class='admin_form'><input type='text' size=40 maxlength=45
name='gpsLongitude' value='$gpsLongitude'></td></tr>");

        if (isset ($_REQUEST['id_to_mod'])) {

            $id_to_mod = $_REQUEST['id_to_mod'];

            echo ("<input TYPE='hidden' NAME='id_to_mod' VALUE='$id_to_mod'
/>");

        }

        echo "<tr><th class='admin_form' nowrap><br>";

        // modify button
        if (isset ($_REQUEST['id_to_mod'])) {
            echo ("<input type='\"submit\"' name='\"action\"' ");
            echo ("value='\"".$_MODIFY."\"' ><br><br> ");
        }

        // add button
    }else{
        echo ("<input type='\"submit\"' name='\"action\"' ");
        echo ("value='\"".$_SAVE."\"' ><br><br> ");
    }

    echo ("</form><form method='\"post\"' action='\"$PHP_SELF\"'>");

    if (isset ($_REQUEST['id_to_mod'])) {

        $id_to_mod = $_REQUEST['id_to_mod'];

        echo ("<input TYPE='hidden' NAME='id_to_mod' VALUE='$id_to_mod'
/>");

    }

    echo ("<input type='\"submit\"' name='\"action\"' ");
    echo ("value='\"".$_CANCEL."\"' >")
    echo ("</td></tr>");
    echo "</table>\n";

    if (isset ($_REQUEST['id_to_mod'])) {

        if ($gpsLatitude != "" AND !is_null($gpsLatitude) AND $gpsLatitude !=
"0.000000" AND $gpsLongitude != "" AND !is_null($gpsLongitude) AND
$gpsLongitude != "0.000000") {

            echo"<div style='height:350px; width:600px' id='test'></div>";

            echo"<script>
                $(document).ready(function(){

```

```

        $('#test').gmap3(
        { action:'init',
        options:{
        center:[$gpsLatitude,$gpsLongitude],
        zoom: 16
        }
        },
        { action: 'addMarker',
        latLng:[$gpsLatitude,$gpsLongitude]
        }
        );
    });
</script>";

    }

}

}

/**
 * Validate input data
 * @access <public>
 *
 *
 * @return <none>
 */

function validate_data($city,$locationName,$address)
{

    if ($city == "" OR is_null($city)) {

        echo "<div class='error'>"._CITY._ISSMISSING."</div>";
        echo ("<br>");

    }

    if ($locationName == "" OR is_null($locationName)) {

        echo "<div class='error'>"._LOCATIONNAME._ISSMISSING."</div>";
        echo ("<br>");

    }

    if ($address == "" OR is_null($address)) {

        echo "<div class='error'>"._ADDRESS._ISSMISSING."</div>";
        echo ("<br>");

    }

}

```

```
}
```

```
/**
 * Add to database
 * @access <public>
 *
 *
 * @return <none>
 */
```

```
function addData($country, $city,$locationName,$address,$zipcode,
$stateProvince,$gpsLongitude,$gpsLatitude) {
```

```
$user_user_id = $_SESSION['verified_user_id'];
```

```
$sql = "INSERT INTO location (country,city,location_name,street_address,zipcode,state_province,gps_longitude,gps_latitude,user_user_id,activity_status_activity_status_id) VALUES('$country', '$city','$locationName','$address','$zipcode','$stateProvince','$gpsLongitude','$gpsLatitude','$user_user_id,1)";
```

```
$result = mysql_query($sql);
```

```
if ($result) {
```

```
header("Location: location.php");
header("Cache-Control: no-store, no-cache, must-revalidate");
header("Cache-Control: post-check=0, pre-check=0", false);
} else {
generatePage();
echo "<div class='error'>".$_ERRORINQUERY."</div>";
echo ("<br><br>");
}
```

```
}
```

```
/**
 * Modify data
 * @access <public>
 *
 *
 * @return <none>
 */
```

```
function modifyData($country, $city,$locationName,$address,$zipcode,
$stateProvince,$gpsLongitude,$gpsLatitude,$id_to_mod) {
```

```
$sql = "UPDATE location SET country = '$country', city = '$city' , location_name = '$locationName', street_address = '$address',zipcode = '$zipcode'
```

```

,state_province = '$stateProvince', _longitude = '$gpsLongitude',gps_latitude
= '$gpsLatitude' WHERE location_id = $id_to_mod";

$result = mysql_query($sql);

if ($result) {

header("Location: location.php");
header("Cache-Control: no-store, no-cache, must-revalidate");
header("Cache-Control: post-check=0, pre-check=0", false);

} else {
generatePage();
echo "<div class='error'>"._ERRORINQUERY."</div>";
echo ("<br><br>");
}

}

/**
 * Delete data, make it inactive
 * @access <public>
 *
 *
 * @return <none>
 */

function deleteData($id_to_mod) {

$sql = "UPDATE location SET activity_status_activity_status_id = 2 WHERE lo-
cation_id = $id_to_mod";

$result = mysql_query($sql);

if ($result) {

header("Location: location.php");
header("Cache-Control: no-store, no-cache, must-revalidate");
header("Cache-Control: post-check=0, pre-check=0", false);

} else {
generatePage();
echo "<div class='error'>"._ERRORINQUERY."</div>";
echo ("<br><br>");
}

}

?>

```

Appendix 2. Android Messenger class to send message to server

```

package fi.toptunniste.android.tagman.connectivity;

import android.util.Log;
import com.google.gson.Gson;
import fi.toptunniste.android.tagman.ServerConnection;
import fi.toptunniste.android.tagman.activity.Tagman;

/**
 *
 * Send data to TagMan server
 *
 */
public class Messenger {

    public Reply sendMessage( Message msg ){

        // Create new server connection
        ServerConnection serverConn = new ServerConnection();

        // Create object of GSON class
        Gson gson = new Gson();

        // Create json formatted string form Message object
        String json = gson.toJson(msg);

        //send data and receive reply
        String reply = serverConn.sendJSONPost(Tagman.SERVER_URL,
        json, msg.getMessageType());

        //Generate Reply object from reply string
        Reply obj = gson.fromJson( reply, Reply.class );

        // Write reply to log file
        Log.v(Tagman.LOG_NAME_FIELD, reply );

        return obj;

    }
}

```

